The basins of attraction of Murakami’s fifth order family of methods

Changbum Chun$^a$, Beny Neta$^{b,*,1}$

$^a$Department of Mathematics, Sungkyunkwan University, Suwon 16419, Republic of Korea
$^b$Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA 93943, United States

A R T I C L E   I N F O

Article history:
Received 10 November 2015
Received in revised form 28 June 2016
Accepted 27 July 2016
Available online 3 August 2016

Keywords:
Iterative methods
Order of convergence
Basin of attraction
Extraneous fixed points

A B S T R A C T

In this paper we analyze Murakami’s family of fifth order methods for the solution of nonlinear equations. We show how to find the best performer by using a measure of closeness of the extraneous fixed points to the imaginary axis. We demonstrate the performance of these members as compared to the two members originally suggested by Murakami. We found several members for which the extraneous fixed points are on the imaginary axis, only one of these has 6 such points (compared to 8 for the other members). We show that this member is the best performer.

Published by Elsevier B.V. on behalf of IMACS.

1. Introduction

There is a vast literature on the solution of nonlinear equations, see for example Ostrowski [19], Traub [23], Neta [16] and Petković et al. [20]. In this paper we consider a fifth-order family of methods and show how to choose the best parameters. We will compare the performance of the two originally suggested members to two new ones by using the idea of basin of attraction and analyzing the extraneous fixed points.

Murakami [15] has developed a fifth order family of methods

\[ x_{n+1} = x_n - a_1 u_n - a_2 w_2(x_n) - a_3 w_3(x_n) - \psi(x_n), \]

where

\[ u_n = \frac{f(x_n)}{f'(x_n)}, \]

\[ w_2(x_n) = \frac{f(x_n)}{f'(x_n - u_n)}. \]

* Corresponding author.

E-mail addresses: cbchun@skku.edu (C. Chun), bneta@nps.edu (B. Neta).

Fax: +1 (831) 656 2355.

http://dx.doi.org/10.1016/j.apnum.2016.07.012
0168-9274/Published by Elsevier B.V. on behalf of IMACS.
\[ w_3(x_n) = \frac{f(x_n)}{f'(x_n + \beta u_n + \gamma w_2(x_n))}, \]

\[ \psi(x_n) = \frac{f(x_n)}{b_1 f'(x_n) + b_2 f'(x_n - u_n)}. \]

This family is of order five when we take

\[ \begin{align*}
    a_1 &= \frac{1}{6} \left( 1 + 4\gamma + \frac{1}{\theta} \right), \\
    a_2 &= \frac{1}{6} \left( \frac{2}{3} \gamma - \frac{1}{3} \right), \\
    a_3 &= \frac{2}{3}, \\
    b_1 &= -\frac{6\theta(\theta - 1)^2}{4\gamma + 1}, \\
    b_2 &= \frac{6\theta^2(\theta - 1)}{4\gamma + 1}, \\
    \beta &= -\gamma - \frac{1}{2}.
\end{align*} \tag{3} \]

and

\[ \theta = \frac{16\gamma + 5}{4(4\gamma + 1)}. \tag{4} \]

Murakami suggested the following two possibilities:

\[ \begin{align*}
    \gamma &= 0, & a_1 &= 0.3, & a_2 &= -0.5, & a_3 &= \frac{2}{3}, \\
    b_1 &= -\frac{15}{32}, & b_2 &= \frac{75}{32}, & \beta &= -\frac{1}{2}
\end{align*} \tag{5} \]

and

\[ \begin{align*}
    \gamma &= -0.5, & a_1 &= -\frac{1}{18}, & a_2 &= -\frac{1}{2}, & a_3 &= \frac{2}{3}, \\
    b_1 &= \frac{9}{32}, & b_2 &= \frac{27}{32}, & \beta &= 0
\end{align*} \tag{6} \]

The idea there probably to choose one of the parameters to be zero, i.e. either \( \gamma = 0 \) or \( \beta = 0 \). As it turns out these parameters are not far from the best.

In this paper, we find the best possible value of the parameter \( \gamma \). We will use two criteria we have developed in previous work \cite{7} based on the location of the extraneous fixed points. In the next section, we discuss the extraneous fixed points. In section \( 3 \) we will discuss the two criteria and give the best parameter based on these criteria. In section \( 4 \) we describe the basins of attraction for the best members of the family for \( 7 \) different examples. We close with conclusions.

2. Extraneous fixed points

For the Murakami family \( z_{n+1} = M_f(z_n) \), where

\[ M_f(z) = z - a_1 u_f(z) - a_2 w_{2,f}(z) - a_3 w_{3,f}(z) - \psi_f(z), \]

\[ u_f(z) = \frac{f(z)}{f'(z)}, \]

\[ w_{2,f}(z) = \frac{f(z)}{f'(z - u_f(z))}, \]

\[ w_{3,f}(z) = \frac{f(z)}{f'(z + \beta u_f(z) + \gamma w_{2,f}(z))}, \]

\[ \psi_f(z) = \frac{f(z)}{b_1 f'(z) + b_2 f'(z - u_f(z))}, \]

we explore its conjugacy on quadratic polynomials. We begin with a preliminary result.

**Lemma 1.** Let \( f(z) \) be an analytic function on the Riemann sphere, and let \( T(z) = \alpha z + \beta, \ \alpha \neq 0 \), be an affine map. If \( g(z) = (f \circ T)(z) \), then we have
Proof. We have \( g'(z) = \alpha f'(T(z)) \), so that

\[
g^{(n)}(z) = \alpha^n f^{(n)}(T(z)), \quad n \geq 1. \tag{9}
\]

From this we obtain

\[
u_f(T(z)) = \frac{f(T(z))}{f'(T(z))} = \alpha u_g(z). \tag{10}\]

We then have by the comparison of Taylor series expansions on \( f \) and \( g \),

\[
f'(T(z) - u_f(T(z))) = f'(T(z) - \alpha u_g(z))
= f'(T(z)) - f''(T(z))\alpha u_g(z) + \cdots 
= \frac{1}{\alpha} g'(z) - \frac{1}{\alpha^2} g''(z)\alpha u_g(z) + \cdots 
= \frac{1}{\alpha} g'(z - u_g(z)), \tag{11}\]

and we obtain

\[
w_{2,f}(T(z)) = \frac{f(T(z))}{f'(T(z) - u_f(T(z)))} = \alpha w_{2,g}(z). \tag{12}\]

Similarly as in (11), it can be shown from (9), (10), (12) that

\[
f'(T(z) + \beta u_f(T(z)) + \gamma w_{2,f}(T(z))) = \frac{1}{\alpha} g'(z + \beta u_g(z) + \gamma w_{2,g}(z)), \tag{13}\]

from which,

\[
w_{3,f}(T(z)) = \alpha w_{3,g}(z). \tag{14}\]

It follows from (9) and (11) that

\[
\psi_f(T(z)) = \alpha \psi_g(z). \tag{15}\]

We can now obtain the scaling theorem for the Murakami family.

**Theorem 2.** Let \( f(z) \) be an analytic function on the Riemann sphere, and let \( T(z) = \alpha z + \beta \). \( \alpha \neq 0 \), be an affine map. If \( g(z) = (f \circ T)(z) \), then we have \( T \circ M_g \circ T^{-1} = M_f(z) \), that is, \( M_f \) and \( M_g \) are topologically conjugated via \( T \).

Proof. We will prove that \( (T \circ M_g)(z) = (M_f \circ T)(z) \) for all \( z \). By Lemma 1 we have

\[
(T \circ M_g)(z) = \alpha M_g(z) + \beta
= \alpha(z - [a_1 u_g(z) + a_2 w_{2,g}(z) + a_3 w_{3,g}(z)] + \alpha \psi_g(z)) + \beta
= T(z) - [a_1 u_f(T(z)) + a_2 w_{2,f}(T(z)) + a_3 w_{3,f}(T(z)) + \psi_f(T(z))]
= (T \circ M_f)(z). \tag{16}\]

Every quadratic polynomial \( p(z) \) with distinct roots reduces, via a linear map, to a polynomial belonging to the one parameter family \( q(z) = z^2 - \mu \), and so by the above scaling theorem, the study of the dynamics of the Murakami family for any complex quadratic polynomial with distinct roots reduces to its dynamical study for \( f(z) = z^2 - \mu \). The scaling theorems and the conjugacy classes of the Murakami family for degree three and four polynomials can be discussed as in Amat et al. [2].

In solving a nonlinear equation iteratively we are looking for fixed points which are zeros of the given nonlinear function. Many multipoint iterative methods have fixed points that are not zeros of the function of interest. Thus, it is imperative to investigate the number of extraneous fixed points, their location and their properties.
Theorem 3. The extraneous fixed points for the Murakami family for \( f(z) = z^2 - \mu \) can be found by solving

\[
\frac{N_f(z)}{D_f(z)} = 0, \tag{17}
\]

where

\[
N_f(z) = (128\gamma^3 + 352\gamma^2 + 522\gamma + 87)z^8 - 2\mu(256\gamma^3 + 224\gamma^2 - 528\gamma - 129)z^6 \\
+ 12\mu^2(64\gamma^3 + 83\gamma + 24)z^4 - 2\mu^2(256\gamma^3 + 32\gamma^2 - 216\gamma - 63)z^2 \\
+ \mu^4(2\gamma + 1)(3 + 8\gamma)^2,
\]

\[
D_f(z) = 6((3 + 16\gamma)z^2 + 16\mu(5 + \gamma))(3 + 2\gamma)z^4 + 4\mu(1 - \gamma)z^2 + \mu^2(1 + 2\gamma)(z^2 + \mu).
\]

Proof. In order to obtain the extraneous fixed points for the method (1), we have to rewrite it in the following form:

\[
x_{n+1} = x_n - u_n H_f(x_n), \tag{18}
\]

where

\[
H_f(x_n) = a_1 + a_2\frac{f'(x_n)}{f'(x_n - u_n)} + a_3\frac{f'(x_n)}{f'(x_n + \beta u_n + \gamma w_2(x_n))}
+ b_1 f'(x_n) + b_2 f'(x_n - u_n) \tag{19}
\]

Upon using \( f(z) = z^2 - \mu \) in (19), we have (17).

In the sequel, the case \( \mu = 1 \) will be considered, so the best possible value of the parameter of the Murakami family will be found for \( f(z) = z^2 - 1 \).

For \( \gamma = 0 \) and \( \mu = 1 \), the equation (17) becomes

\[
\frac{1}{2} \frac{29z^6 + 57z^4 + 39z^2 + 3}{(z^2 + 1)(3z^2 + 5)(3z^2 + 1)} = 0. \tag{20}
\]

The roots are as follows;

\[
\xi = \pm 0.296058904382937i, \quad \xi = \pm 0.271524554061638 \pm 1.00630947859782i.
\]

All the fixed points are repulsive.

For \( \gamma = -0.5 \) and \( \mu = 1 \), the equation (17) becomes

\[
\frac{N_f(z)}{(z^2 + 3)(z^2 + 1)(z^2 + 0.6)} = 0, \tag{21}
\]

where

\[
N_f(z) = 1.7(z^2 + 0.467734312641029z + 1.58640402523038) \\
(z^2 + 0.163614395584542)(z^2 - 0.467734312641029z + 1.58640402523038).
\]

The roots are as follows;

\[
\xi = \pm 0.4044927633i, \quad \xi = \pm 0.2338671563 \pm 1.237622793i.
\]

All the fixed points are repulsive.

In the next section we find that, based on the criterion used, the parameter \( \gamma \) should be either \( \gamma = -0.62 \) or \( \gamma = -0.125 \).

For \( \gamma = -0.62 \) and \( \mu = 1 \), the equation (17) becomes

\[
\frac{N_f(z)}{D_f(z)} = 0, \tag{22}
\]

where

\[
N_f(z) = 1.804130321(z^2 + 0.698167010067880z + 1.89250439483774) \\
(z^2 + 0.0593972226948352)(z^2 + 0.032873468882183) \\
(z^2 - 0.698167010067880z + 1.89250439483774),
\]

\[
D_f(z) = (z^2 + 0.7109826589)(z + 0.191498766272523)(z - 0.191498766272523) \\
(z^2 + 3.71848995930208)(z^2 + 1).
\]
The roots are as follows;

$$\xi = \pm 0.3490835050 \pm 1.330655891i, \xi = \pm 0.2437154544i, \xi = \pm 0.1813104213i.$$ 

All the fixed points are repulsive.

For $\gamma = -0.125$ and $\mu = 1$, the equation (17) becomes

$$\frac{N_\gamma(z)}{D_\gamma(z)} = 0,$$

where

$$N_\gamma(z) = 1.636363636(z^2 + 0.0463327506379597)(z^2 + 1)(z^2 + 2.39811169380649),$$

$$D_\gamma(z) = (z^2 + 1.44801847547959)(z^2 + 3)(z^2 + 0.188345160884045).$$

The roots are as follows;

$$\xi = \pm 1.548583770i, \xi = \pm i, \xi = \pm 0.2152504370i.$$ 

All the fixed points are repulsive.

**Remark.** Notice that this is the only case out of the 4 above where the extraneous fixed points are all on the imaginary axis. There are other values of $\gamma$ in the range $(-0.1891, -0.125)$ for which we have purely imaginary extraneous fixed points. In these cases the number of points is 8 versus 6 in the case we chose to present here.

In the next section we discuss two criteria to choose the parameters in the Murakami family. These criteria were developed in [8].

### 3. Best possible parameters

The parameters can be chosen to position the extraneous fixed points on the imaginary axis or, at least, close to that axis, (see, for example, Chun and Neta [7] and [10]).

We have searched the parameter space ($\gamma$) and found that the extraneous fixed points are on the imaginary axis for certain values of the parameter $\gamma$. We have considered one measure of closeness to the imaginary axis, denoted by $d$, and another measure of averaged stability of the extraneous fixed points, denoted by $A$. These measures are defined below. We have experimented with those members from the parameter space.

Let $E = \{z_1, z_2, \ldots, z_{n_E}\}$ be the set of the extraneous fixed points corresponding to the value of the parameter $\gamma$. We define

$$d(\gamma) = \max_{z_i \in E} |\text{Re}(z_i)|.$$  

We look for the parameters $\gamma$ which attain the minimum of $d(\gamma)$. The minimum of $d(\gamma) = 0$ occurs at $\gamma = -0.125$. We will call the corresponding method **MurakamiA**. We will not consider other values of $\gamma$ for which $d(\gamma) = 0$, since they have more extraneous fixed points.

Another method to choose the parameter is by considering the stability of $z \in E$ defined by

$$dq(z) = \frac{d q}{d z}(z),$$

where $q$ is the iteration function of the Murakami family of methods. We define a function called the averaged stability value of the set $E$ by

$$A(\gamma) = \frac{\sum_{z_i \in E} |dq(z_i)|}{n_\gamma},$$

where $n_\gamma$ is the number of elements of the set $E$.

The smaller $A$ becomes, the less chaotic the basin of attraction tends to. The minimum of $A(\gamma)$ occurs at $\gamma = -0.62$. We will call the corresponding method **MurakamiA**.

It was shown before that methods for which the extraneous fixed points are on or close to the imaginary axis perform better than others, $d(\gamma)$ measures the distance of the extraneous fixed points from the imaginary axis and thus predicts best performance. The measures $d(\gamma)$ and $A(\gamma)$ provide ways of choosing the best performers for any parameter-dependent families of methods. In our previous work, we found that methods suggested by these measures perform better than others.

The values of the parameter $\gamma$ presented in [15] are $\gamma = 0$ and $\gamma = -0.5$ that we call them **Murakami1** and **Murakami2**, respectively.
4. Numerical experiments

The Basin of Attraction is a method to visually understand how a method behaves as a function of the various starting points. This idea was started by Stewart [22] and continued in the work of Amat et al. [1–3], Argyros and Magreñán [4], Chicharro et al. [5], Chun et al. [6,9,11], Cordero et al. [12], Geum et al. [13], Magreñán [14], Neta et al. [17,18], and Scott et al. [21].

We have used the 4 members of the Murakami family for 7 different polynomials. The choice of the parameters in the families used is based on the analysis in the previous section. All the examples have roots within a square of $[-3, 3]$ by $[-3, 3]$. We have taken 360,000 equally spaced points in the square as initial points for the methods and we have registered the total number of iterations required to converge to a root and also to which root it converged. We have also collected the CPU time (in seconds) required to run each method on all the points using Dell Optiplex 990 desktop computer. We then computed the average number of iterations required per point and the number of points requiring 40 iterations.

Example 1. In our first example, we have taken the polynomial
\[ p_1(z) = z^2 - 1 \] (27)
whose roots are $z = \pm 1$. In Fig. 1 we have presented the basins for the 4 members of the Murakami family of methods. In the top row, we have $\gamma = 0$ (left) and $\gamma = -0.5$ (right). On the bottom row we have the case $\gamma = -0.62$ based on the $A$ criterion (left) and $\gamma = -0.125$ based on the $d$ criterion (right). It is clear from the Fig. 1 that Murakamid is the best method (bottom right subplot). The parameter based on the $d$ criterion for closeness has improved the basins (compared to the cases suggested by [15]). The parameter based on the $A$ criterion made the situation slightly worse. In order to get a quantitative comparison, we have collected the average number of iterations per point in Table 1, the CPU time for each method and each example in Table 2 and the number of points requiring 40 iterations in Table 3. It is clear that Murakamid requires the least number (2.70), and the original ones require slightly more. Murakamia is the worst, requiring 2.94 iterations per point on average. Note that the difference is very small. In terms of CPU time (see Table 2) now Murakami2 is the fastest (150 seconds) followed by Murakamid (151 seconds). Murakamia is the slowest with 162 seconds. In terms of the number of points requiring 40 iterations (see Table 3) we find that MurakamiA is the worst with 765 points requiring 40 iterations. All other methods have 601 points requiring 40 iterations.

Example 2. Our next example is a cubic polynomial having the three roots of unity,
Table 1
Average number of iterations per point for each example (1–7) and each method.

<table>
<thead>
<tr>
<th>Method</th>
<th>Ex1</th>
<th>Ex2</th>
<th>Ex3</th>
<th>Ex4</th>
<th>Ex5</th>
<th>Ex6</th>
<th>Ex7</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Murakami1 ($\gamma = 0$)</td>
<td>2.81</td>
<td>3.43</td>
<td>3.52</td>
<td>2.98</td>
<td>4.64</td>
<td>4.21</td>
<td>5.93</td>
<td>3.93</td>
</tr>
<tr>
<td>Murakami2 ($\gamma = -0.5$)</td>
<td>2.72</td>
<td>3.39</td>
<td>3.38</td>
<td>2.92</td>
<td>4.62</td>
<td>4.24</td>
<td>6.06</td>
<td>3.90</td>
</tr>
<tr>
<td>MurakamiA ($\gamma = -0.62$)</td>
<td>2.94</td>
<td>3.58</td>
<td>3.47</td>
<td>3.09</td>
<td>4.84</td>
<td>4.48</td>
<td>6.35</td>
<td>4.11</td>
</tr>
<tr>
<td>Murakamid ($\gamma = -0.125$)</td>
<td>2.70</td>
<td>3.36</td>
<td>3.29</td>
<td>2.88</td>
<td>4.34</td>
<td>3.63</td>
<td>5.09</td>
<td>3.61</td>
</tr>
</tbody>
</table>

Table 2
CPU time (in seconds) required for each example (1–7) and each method.

<table>
<thead>
<tr>
<th>Method</th>
<th>Ex1</th>
<th>Ex2</th>
<th>Ex3</th>
<th>Ex4</th>
<th>Ex5</th>
<th>Ex6</th>
<th>Ex7</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Murakami1</td>
<td>156.297</td>
<td>252.331</td>
<td>258.228</td>
<td>289.428</td>
<td>417.116</td>
<td>1297.023</td>
<td>648.995</td>
<td>474.203</td>
</tr>
<tr>
<td>Murakami2</td>
<td>150.338</td>
<td>248.837</td>
<td>240.382</td>
<td>278.508</td>
<td>420.501</td>
<td>1296.010</td>
<td>657.076</td>
<td>470.236</td>
</tr>
<tr>
<td>MurakamiA</td>
<td>162.584</td>
<td>259.336</td>
<td>244.734</td>
<td>295.357</td>
<td>438.565</td>
<td>1358.800</td>
<td>679.961</td>
<td>491.334</td>
</tr>
<tr>
<td>Murakamid</td>
<td>151.024</td>
<td>244.407</td>
<td>244.235</td>
<td>276.402</td>
<td>386.477</td>
<td>1111.538</td>
<td>572.367</td>
<td>426.636</td>
</tr>
</tbody>
</table>

Table 3
Number of points requiring 40 iterations for each example (1–7) and each method.

<table>
<thead>
<tr>
<th>Method</th>
<th>Ex1</th>
<th>Ex2</th>
<th>Ex3</th>
<th>Ex4</th>
<th>Ex5</th>
<th>Ex6</th>
<th>Ex7</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Murakami1</td>
<td>601</td>
<td>1</td>
<td>0</td>
<td>601</td>
<td>19</td>
<td>0</td>
<td>1027</td>
<td>321.29</td>
</tr>
<tr>
<td>Murakami2</td>
<td>601</td>
<td>1</td>
<td>0</td>
<td>601</td>
<td>20</td>
<td>0</td>
<td>886</td>
<td>301.29</td>
</tr>
<tr>
<td>MurakamiA</td>
<td>765</td>
<td>1</td>
<td>0</td>
<td>601</td>
<td>41</td>
<td>1</td>
<td>948</td>
<td>336.71</td>
</tr>
<tr>
<td>Murakamid</td>
<td>601</td>
<td>2086</td>
<td>4</td>
<td>609</td>
<td>5483</td>
<td>28</td>
<td>6158</td>
<td>2138.43</td>
</tr>
</tbody>
</table>

Fig. 2. The top row for Murakami1 (left) and Murakami2 (right), and second row for MurakamiA (left) and Murakamid (right) for the roots of the polynomial $z^3 - 1$.

\[ p_2(z) = z^3 - 1. \]  

(28)

The basins of attraction are plotted in Fig. 2. Again MurakamiA is most chaotic and the others are about the same. The average number of iterations per point is lowest for Murakamid (3.36) and highest for MurakamiA (3.58). The CPU time
Fig. 3. The top row for Murakami1 (left) and Murakami2 (right), and second row for MurakamiA (left) and Murakamid (right) for the roots of the polynomial $z^3 - z$.

is about the same for all (249–259 seconds) with MurakamiA being the slowest and Murakamid being the fastest. The methods with the fewest number of points requiring 40 iterations are: Murakami1, Murakami2 and MurakamiA (all with one point). The worst is Murakamid with 2086 points.

**Example 3.** Our next example is the cubic polynomial,

$$p_3(z) = z^3 - z.$$  \hspace{1cm} (29)

The basins of attraction are given in Fig. 3. It is clear that Murakamid is best, but Murakami1 and Murakami2 are not far behind. In terms of the average number of iterations per points, Murakamid is best (3.29) and Murakami1 is worst (3.52). The fastest is Murakami2 (240 seconds) and the slowest is Murakami1 (258 seconds). All members have no black points (see Table 3) except Murakamid with 4 black points.

**Example 4.** The fourth example is the quartic polynomial,

$$p_4(z) = z^4 - 10z^2 + 9.$$ \hspace{1cm} (30)

The basins of attraction are plotted in Fig. 4. Again Murakamid is better than the others. This is also confirmed by consulting Tables 1 and 2. All members have the same numbers of points requiring 40 iterations (601 points) except Murakamid with 609 points.

**Example 5.** The fifth example is a polynomial

$$p_5(z) = z^5 - 1.$$ \hspace{1cm} (31)

The plots of the basins are given in Fig. 5. Murakamid and Murakami1 are better than the others. The average number of iterations per point is 4.34 for Murakamid and higher (4.62–4.84) for the others (see Table 1). Murakamid is the fastest with 386 seconds (see Table 2) followed by Murakami1 (417 seconds) and Murakami2 (420 seconds). In terms of the number of points requiring 40 iterations, Murakami1 and Murakami2 have fewest number (19–20 points) and Murakamid the highest with 5483 points.
Fig. 4. The top row for Murakami1 (left) and Murakami2 (right), and second row for MurakamiA (left) and MurakamiB (right) for the roots of the polynomial $z^4 - 10z^2 + 9$.

Fig. 5. The top row for Murakami1 (left) and Murakami2 (right), and second row for MurakamiA (left) and MurakamiB (right) for the roots of the polynomial $z^5 - 1$. 
Example 6. Our next example is a polynomial with complex coefficients

\[ p_6(z) = z^6 - \frac{1}{2} z^5 + \frac{11}{4} (1 + i) z^4 - \frac{1}{4} (19 + 3i) z^3 + \frac{1}{4} (11 + 5i) z^2 - \frac{1}{4} (11 + i) z + \frac{3}{2} - 3i. \]  

(32)

This example was the hardest for many iterative methods as we found out in our previous work. Murakamid is the best, as can be seen from Fig. 6. From Table 1 we find that Murakamid requires 3.63 iterations per point on average and the others require above 4.21. Murakamid is the fastest with 1111 seconds, followed by Murakami2 (1296 seconds) and Murakami1 (1297 seconds). The number of points requiring 40 iterations (see Table 3) is highest for Murakamid with 28 points. The rest have no black point or just one point.

Example 7. Our last example is a polynomial

\[ p_7(z) = z^7 - 1. \]  

(33)

The basins of attraction are plotted in Fig. 7. Based on the figure we conclude that Murakamid is best. Based on Table 1, we conclude that Murakamid uses least number of iterations per point (5.09) and the others use 5.93–6.35 iterations per point on average. The fastest method (see Table 2) is Murakamid (572 seconds) and the slowest is MurakamiA with 680 seconds. The number of points requiring 40 iterations is the smallest for Murakami2 (886) and the highest (6158) for Murakamid.

5. Conclusions

In order to decide which member is best overall, we have averaged the numbers across the examples. We now find that Murakamid requires the least number of iterations per point (3.61) followed by Murakami2 (3.90). MurakamiA requires the most (4.11). Same conclusion for the average CPU time across examples, i.e. Murakamid is the fastest (426 seconds) and MurakamiA is the slowest (491 seconds). In terms of the number of points requiring 40 iterations, the highest is 2138 for Murakamid and the lowest (301) for Murakami2. We can conclude that the criterion \( d \) is very useful in finding the best performer from a family of methods. Based on the averages in the 3 tables we conclude that Murakamid is best followed by Murakami2 and Murakami1.
Fig. 7. The top row for Murakami1 (left) and Murakami2 (right), and second row for MurakamiA (left) and MurakamiD (right) for the roots of the polynomial $z^2 - 1$.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013R1A1A2005012). The first author thanks the Applied Mathematics Department at the Naval Postgraduate School for hosting him during the years.

References