Bisheh-Niasar–Saadatmandi Root Finding Method via the S-iteration with Periodic Parameters and Its Polynomiography

Morteza Bisheh-Niasar[a,*], Krzysztof Gdawiec[b]

[a] *Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Kashan, Kashan, Iran*
[b] *Institute of Computer Science, University of Silesia, Będzińska 39, 41–200 Sosnowiec, Poland*

Abstract

In recent years many researchers have focused their attention on the use of different iteration process – known from fixed point theory – in the generation of different kinds of patterns. In this paper, we propose modifications of the Saadatmandi and Bisheh-Niasar root finding method. In the first modification we modify the formula of the method and in the second modification we use the S-iteration with periodic parameters. Moreover, we numerically investigate some properties of the proposed methods and modification using three measures, i.e., the generation time, mean number of iterations and convergence area index. The obtained polynomiographs show that the proposed methods have a potential artistic applications, and the numerical results show that there is no obvious dependency of the considered measures on the sequences of the parameters used in the S-iteration.

Keywords: S-iteration, polynomiography, root finding

1. Introduction

Using the computer in many of industrial and economic activities, is undeniable. As nowadays, in order to promote variety of products, using of some hardware, software and technical innovation in competition, is needed. In textile and carpet industries, patterns generation can be one instance of applications of computer. In carpet design and tapestry design, a designer have to be aware of these techniques, in order to obtain an interesting pattern. Therefore, it is highly motivated to develop new methods of obtaining an interesting patterns. Polynomials along with the roots finding methods can be used for the generation of these.

*Corresponding author
Email addresses: mbisheh@kashanu.ac.ir (Morteza Bisheh-Niasar), kgdawiec@ux2.math.us.edu.pl (Krzysztof Gdawiec)

Polynomial root finding is one of the oldest mathematical problems. Historical documents show that Sumerian (3000 B.C.) and Babylonians (2000 B.C.) dealt with it. In 17th century, Newton proposed a numerical method for approximation of roots of polynomials. Cayley, in 1879, investigated the behaviour of Newton's method for equation $z^3 - 1 = 0$ in the complex plane (which is known as the Cayley's problem). Finally in 1919, Cayley's problem was solved by Julia.

In connection with polynomial roots finding, Kalantari introduced an interesting subject the so-called "Polynomiography". Polynomiography is defined to be "the art and science of visualization in approximation of the zeros of complex polynomials, via fractal and non-fractal images created using the mathematical convergence properties of iteration functions" [1, 2]. An individual image created with the mentioned methods and properties is called a "polynomiograph".

Kalantari used polynomiography not only for scientific aims, but also as an inspiration for artworks [3], e.g., paintings, carpet design, tapestry, sculptures. The artistic applications of polynomiography drawn attention of many scientists. They proposed various modifications of the methods proposed by Kalantari. The most popular modification relays on the use of various iteration processes from fixed point theory. In [4] the authors have proposed the use of Mann and Ishikawa iteration. Later, in [5] the S-iteration was used. Eleven different iteration processes were studied in [6]. In [7] Rafiq et al., instead of studying the explicit iteration processes as the previous researchers, started to study the implicit iteration processes. They used Jungck-Mann and Jungck-Ishikawa iterations. All the iteration processes use parameters, which are sequences. In all the mentioned papers for simplicity the authors used constant sequences of the parameters. The iteration processes were not only studied as the source for new artistic patterns, but also in the computational aspect. In [8, 9] the authors made a comparison of different iteration process in root finding using two measures: the mean number of iterations, convergence area index.

In this paper, we propose modifications of the root finding method introduced in [10]. In the first modification we modify the formula of this method and the second modification relays on the use of the S-iteration. Moreover, instead of using constant sequences in the S-iteration we propose the use of sequences that are periodic functions. The proposed methods are studied in two directions – artistic and computational.

This paper is organized as follows: In Sec. 2, we introduce the basics of polynomiography. Next, in Sec. 3 we present root finding method introduced in [10]. Moreover, we propose modification of the S-iteration process by introducing parameters defined with periodic functions. Some graphical examples and numerical results are presented in Sec. 4. Finally, in Sec. 5 we give some concluding remarks.

2. Polynomiography

As it was already mentioned in the introduction polynomiography is a method of visualizing the approximation of zeroes of complex polynomials. Thus, the main element in the polynomiography – except the polynomial – is a method for the approximation of zeroes, i.e., the root finding method. The literature is full of various root finding methods. The oldest and the most well-known method is the Newton's root finding method. It was introduced in the 17th century and since then many different methods were proposed, e.g., Halley method [11], Traub-Ostrowski method [12], Harmonic Mean method [12], or even whole families of methods, e.g., Basic family [2], Euler-Schröder family [2], Jarratt family [13].

To generate a polynomiograph we take area of interest in the complex plane $A \subset \mathbb{C}$ and for each point z_0 in this area we iterate the root finding method R using a feedback iteration, i.e.,

$$z_{n+1} = R(z_n). \tag{1}$$

We proceed with the iteration process till the convergence test is satisfied or the maximum number of iterations is reached. The standard convergence test has the following form:

$$|z_{n+1} - z_n| < \varepsilon, \tag{2}$$

where $\varepsilon > 0$ is the accuracy of the computations. Finally, when we end the iteration process we colour the starting point (z_0) using some colouring function. The two basic colouring functions are: colouring according to the iteration (we assign the colour using the number of performed iterations and a colour map), basins of attraction (each root of the polynomial gets its own distinct colour and we assign the colour using the colour of the nearest root to the point at which we have stopped iterating).

The method of generating polynomiograph described above is the standard way introduced by Kalantari. In the literature exist various modifications of this method. The two most commonly used modifications relay on the use of different iteration processes and different convergence tests.

In the first modification we replace the feedback iteration process (1) (the Picard iteration) with other iteration processes known from fixed point theory. In the literature various iterations processes were used, e.g., Mann, Ishikawa, S, Noor, Jungck-Mann, Jungck-Ishikawa. Lets recall some of the processes:

- Mann iteration [14]

  $$z_{n+1} = (1 - \alpha_n) z_n + \alpha_n R(z_n), \quad n = 0, 1, 2, \ldots, \tag{3}$$

  where $\alpha_n \in (0, 1]$ for all $n \in \mathbb{N}$.

- Ishikawa iteration [15]

  $$\begin{cases} z_{n+1} = (1 - \alpha_n) z_n + \alpha_n R(u_n), \\ u_n = (1 - \beta_n) z_n + \beta_n R(z_n), \quad n = 0, 1, 2, \ldots, \end{cases} \tag{4}$$

  where $\alpha_n \in (0, 1]$ and $\beta_n \in [0, 1]$ for all $n \in \mathbb{N}$.

- S-iteration [16]

$$\begin{cases} z_{n+1} = (1-\alpha_n)R(z_n) + \alpha_n R(u_n), \\ u_n = (1-\beta_n)z_n + \beta_n R(z_n), \quad n = 0,1,2,\ldots, \end{cases} \tag{5}$$

where $\alpha_n \in (0,1]$ and $\beta_n \in [0,1]$ for all $n \in \mathbb{N}$.

- Noor iteration [17]

$$\begin{cases} z_{n+1} = (1-\alpha_n)z_n + \alpha_n R(u_n), \\ u_n = (1-\beta_n)z_n + \beta_n R(v_n), \\ v_n = (1-\gamma_n)z_n + \gamma_n R(z_n), \quad n = 0,1,2,\ldots, \end{cases} \tag{6}$$

where $\alpha_n \in (0,1]$ and $\beta_n, \gamma_n \in [0,1]$ for all $n \in \mathbb{N}$.

One can easily observe that the iterations for particular values of the parameters can be reduced to other iterations, e.g., Ishikawa iteration with $\beta_n = 0$ for $n \in \mathbb{N}$ is a Mann iteration, and when $\beta_n = 0$, $\alpha_n = 1$ for $n \in \mathbb{N}$ is the Picard iteration. A review of 18 different iteration processes and their dependencies can be found in [18].

In the second modification – presented in [19] – the standard convergence test (2) is replaced by other convergence tests. The tests are based not only on metrics, but also on functions that are not metrics. Using those new tests we are able to introduce a very interesting details to polynomiographs. An exemplary convergence tests that can be used are the following:

$$|0.01(z_{i+1} - z_i)| + |0.029|z_{i+1}|^2 - 0.03|z_i|^2| < \varepsilon, \tag{7}$$

$$\left| \frac{0.05}{|z_{n+1}|^2} - \frac{0.05}{|z_n|^2} \right| < \varepsilon, \tag{8}$$

$$|0.04\Re(z_{i+1} - z_i)| < \varepsilon \vee |0.05\Im(z_{i+1} - z_i)| < \varepsilon, \tag{9}$$

where $\Re(z)$, $\Im(z)$ is the real and imaginary part of z, respectively.

The pseudocode of the algorithm for the generation of polynomiograph using iteration processes and convergence tests is presented in Algorithm 1.

3. Root finding method and its modifications

In [10] Saadatmandi and Bisheh-Niasar introduced a new root finding method. They used Taylor's expansion

$$f(\alpha) = f(x_n) + (\alpha - x_n)\frac{f'(x_n)}{1!} + (\alpha - x_n)^2\frac{f''(x_n)}{2!} + \ldots + (\alpha - x_n)^k\frac{f^{(k)}(x_n)}{k!} + \ldots, \tag{10}$$

where α is a root, and the substitution of $\alpha - x_n$ with

$$\frac{\exp(b(\alpha - x_n)) - 1}{b}, \tag{11}$$

Algorithm 1: Polynomiograph generation

Input: $p \in \mathbb{C}[Z]$ – polynomial, $A \subset \mathbb{C}$ – area, K – maximum number of iterations, I_q – iteration method, $q \in \mathbb{C}^N$ – parameters of the iteration I_q, R – root finding method, T_t – convergence test, $t \in \mathbb{R}^M$ – parameters of the convergence test T_t.

Output: Polynomiograph for the area A.

for $z_0 \in A$ **do**
    $n = 0$
    while $n \leq K$ **do**
        $z_{n+1} = I_q(R, p, z_n)$
        if $T_t(z_n, z_{n+1}) = true$ **then**
            break
        $n = n + 1$
    determine the colour for z_0

where $b = f''(x_n)/f'(x_n)$ to obtain the following method:

$$z_{n+1} = z_n + \frac{f'(z_n)}{f''(z_n)} \ln\left(1 - \frac{f(z_n)f''(z_n)}{f'(z_n)^2}\right). \tag{12}$$

This method has the third-order convergence [10].

Now, introducing

$$R(z) = z + \frac{f'(z)}{f''(z)} \ln\left(1 - \frac{f(z)f''(z)}{f'(z)^2}\right) \tag{13}$$

we can write (12) in the following short form:

$$z_{n+1} = R(z_n). \tag{14}$$

In [6] the authors used various iteration methods in the root finding methods. In the rest of the paper we will be considering (13) with only one iteration method, namely the S-iteration, i.e.,

$$\begin{cases} z_{n+1} = (1 - \alpha_n)R(z_n) + \alpha_n R(u_n), \\ u_n = (1 - \beta_n)z_n + \beta_n R(z_n), \quad n = 0, 1, 2, \ldots, \end{cases} \tag{15}$$

where $\alpha_n \in (0, 1]$ and $\beta_n \in [0, 1]$ for all $n \in \mathbb{N}$.

To obtain more interesting polynomiographs, from the artistic point of view, we modify (13) to the following form:

$$S(z) = \frac{f'(z)}{f''(z)} \ln\left(1 - \frac{f(z)f''(z)}{f'(z)^2}\right) \tag{16}$$

and use it with the S-iteration:

$$\begin{cases} z_{n+1} = (1-\alpha_n)S(z_n) + \alpha_n S(u_n), \\ u_n = (1-\beta_n)z_n + \beta_n S(z_n), \quad n = 0, 1, 2, \ldots, \end{cases} \tag{17}$$

where $\alpha_n \in (0, 1]$ and $\beta_n \in [0, 1]$ for all $n \in \mathbb{N}$. This method usually does not converge (using the standard convergence test) to the roots of a given polynomial, but as we will see in the examples presented in Sec. 4 the patterns obtained with this method are more interesting than the ones generated by (13).

In the previous papers on polynomiography [18, 6, 5] only constant parameters (α_n, β_n) in the S-iteration were used. In this paper we propose the use of parameters that are defined with the use of periodic functions. For instance, as the periodic functions one can take the trigonometric functions. This modification will allow to change the dynamics of the root finding method in a significant way. This change of the dynamics is other than when we use the constant values of the parameters. Moreover, we extend the intervals of the possible values of the parameters from $[0, 1]$ to $[-1, 1]$.

4. Examples

In this section, we present some polynomiographs obtained with the methods introduced in Sec. 3. Moreover, we compare (13) and (16) using three measures: generation time, mean number of iterations and convergence area index.

In all the examples the common parameters used to generate the polynomiographs were the following: $p(z) = z^5 - 1$, $K = 10$, resolution of 301×301 pixels. The examples are divided according to the functions that were used for defining the parameters of the S-iteration. Moreover, in each example we use two convergence tests, namely test (2) and (8) with $\varepsilon = 0.001$.

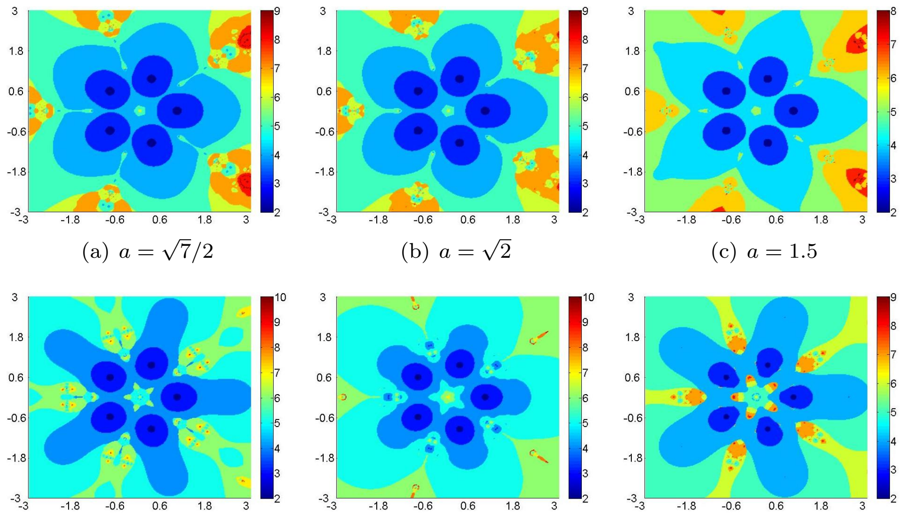
The three measures used in the comparison are defined as follows. The generation time is the time needed to compute all the points of the polynomiograph. It is measured in seconds. The mean number of iterations (MNI) is computed – from the polynomiograph obtained with the colouring according to the iteration – as the mean value of the iterations in the polynomiograph. The convergence area index (CAI) is given by the following formula:

$$CAI = \frac{N_c}{N}, \tag{18}$$

where N_c is the number of points in the polynomiograph that have converged and N is the number of all points in the polynomiograph. The value of CAI is between 0 (no point has converged) and 1 (all points have converged).

The algorithm for the polynomiographs' generation has been implemented in Matlab and all polynomiographs have been generated on a computer with the following specification: Intel i3-4130 (@3.4 GHz) processor, 4 GB RAM and Windows 8 (64-bit).

In the first example we use the following sequences of parameters $\alpha_n = \cos(an)$ and $\beta_n = \sin(an)$, where $a \in \mathbb{R}$ is the parameter of the sequence in



(a) $a = \sqrt{7}/2$ (b) $a = \sqrt{2}$ (c) $a = 1.5$

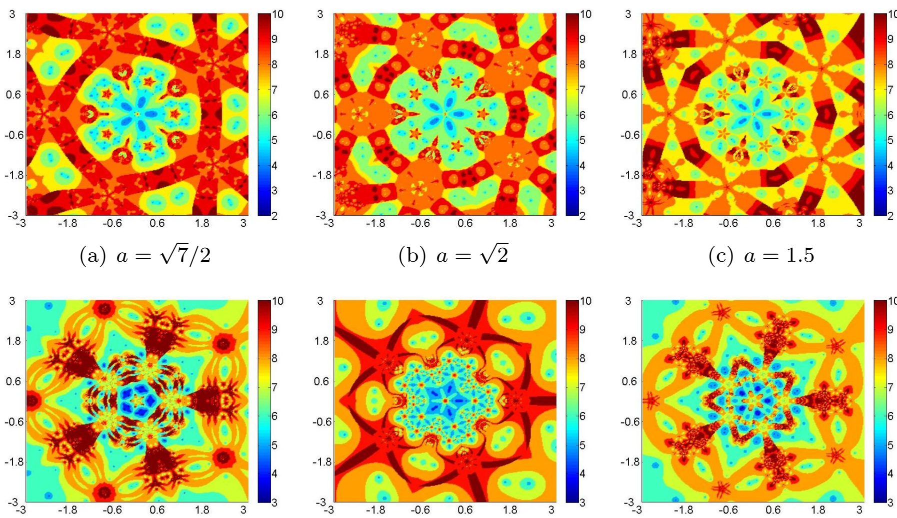
(d) $a = 2$ (e) $a = 2.5$ (f) $a = 5$

Figure 1: Polynomiographs generated using (13), the convergence test (2) and $\alpha_n = \cos(an), \beta_n = \sin(an)$ in the S-iteration

the S-iteration. Changing the a parameter we change the periodicity of the sequence. The polynomiographs generated in $A = [-3, 3]^2$ using (13) and (16) are presented in Fig. 1 and 2, respectively. To generate these polynomiographs the standard convergence test (2) was used. Looking at the polynomiographs in both figures we see that the change of the periodicity of the sequences α_n, β_n has a great influence on the shape of the polynomiograph, especially in the case of using (16). Moreover, the patterns obtained with (13) have a very regular and smooth shape, whereas the patterns generated using (16) have irregular shapes and possess much more details, so they are much more interesting from the artistic point of view.

The numerical results obtained for the polynomiographs in Fig. 1 and 2 are presented in Tab. 1. In the case of the method (13) we see that the generation times are between 32 and 35 seconds, attaining the minimum at $a = 1.5$ and the maximum at $a = \sqrt{2}$. The generation times for (16) are longer, ranging from 55 to 59 seconds. This difference is caused by the fact that the MNI for (16) has higher values than in the case of (13). The MNI is about 1.6-1.7 times higher, so we need to make more computations. For the CAI measure we see that in the case of (13) the change of the periodicity has a small effect on its value, whereas for (16) the CAI measure changes in a significant way.

The polynomiographs obtained with the same methods and α_n, β_n in the S-iteration, but with the convergence test (8) are presented in Fig. 3 and 4. The use of other convergence test in both cases introduced new details into the polynomiographs. For the method (13) the new details are visible in the regions where the method has converged fast using the standard convergence test. The



(a) $a = \sqrt{7}/2$ (b) $a = \sqrt{2}$ (c) $a = 1.5$

(d) $a = 2$ (e) $a = 2.5$ (f) $a = 5$

Figure 2: Polynomiographs generated using (16), the convergence test (2) and $\alpha_n = \cos(an), \beta_n = \sin(an)$ in the S-iteration

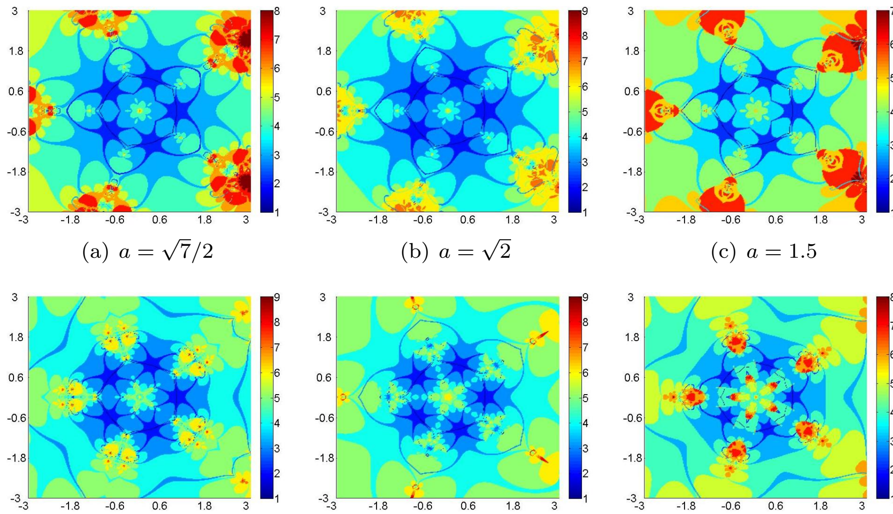
Table 1: Results obtained for the convergence test (2) and $\alpha_n = \cos(an), \beta_n = \sin(an)$ in the S-iteration

(a) method (13)

a	Time	MNI	CAI
$\sqrt{7}/2$	33.7657	4.6507	1
$\sqrt{2}$	35.1831	4.6971	1
1.5	32.0688	4.4909	1
2	33.9159	4.7258	0.9999
2.5	34.0460	4.8652	0.9999
5	33.1420	4.6792	1

(b) method (16)

a	Time	MNI	CAI
$\sqrt{7}/2$	59.3970	7.741	0.9406
$\sqrt{2}$	58.2912	7.5479	0.9751
1.5	59.4442	7.6733	0.8827
2	55.3757	7.5744	0.8722
2.5	59.8557	7.9096	0.8949
5	55.7959	7.4452	0.9496



(a) $a = \sqrt{7}/2$ (b) $a = \sqrt{2}$ (c) $a = 1.5$

(d) $a = 2$ (e) $a = 2.5$ (f) $a = 5$

Figure 3: Polynomiographs generated using (13), the convergence test (8) and $\alpha_n = \cos(an), \beta_n = \sin(an)$ in the S-iteration

obtained patterns look more interesting. In the case of (16) the new details are not as visible as in the case of (13), but nevertheless they make the patterns look more intriguing.

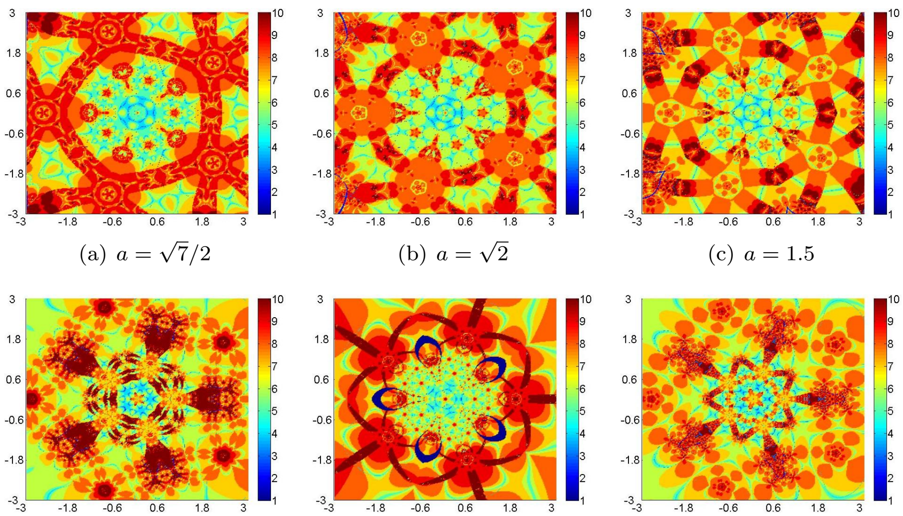
Tab. 2 gathers the results of numerical computations regarding the polynomiographs from Fig. 3 and 4. The results show that the use of other convergence test has only a small impact on the generation time. The value of the MNI in both cases has decreased. For the (13) from about 0.5 to 0.6, whereas from 0.2 to 0.4 for the (16). Moreover, we see that the change of the periodicity for (13) does not affect the CAI, because for all the values of a the CAI is equal 1. Thus, for all points the method has converged to the roots. In the second case (method (16)) we see that for all values of a the value of CAI is less than 1, so in the considered area there are some points that do not converge to the roots using the second convergence test.

For the next examples let us consider the following functions:

$$f(x) = \begin{cases} -2(1+x) & \text{if } -1 \leq x \leq -0.5, \\ 2x & \text{if } -0.5 < x < 0.5, \\ 2(1-x) & \text{if } 0.5 \leq x \leq 1, \end{cases} \tag{19}$$

$$g(x) = -x, \quad \text{for } x \in (-1, 1). \tag{20}$$

Assume that f and g have the period $T = 2$. Now we define α_n and β_n based



(a) $a = \sqrt{7}/2$ (b) $a = \sqrt{2}$ (c) $a = 1.5$

(d) $a = 2$ (e) $a = 2.5$ (f) $a = 5$

Figure 4: Polynomiographs generated using (16), the convergence test (8) and $\alpha_n = \cos(an), \beta_n = \sin(an)$ in the S-iteration

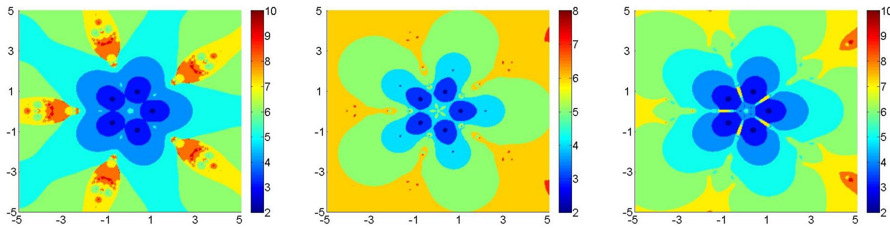
Table 2: Results obtained for the convergence test (8) and $\alpha_n = \cos(an), \beta_n = \sin(an)$ in the S-iteration

(a) method (13)

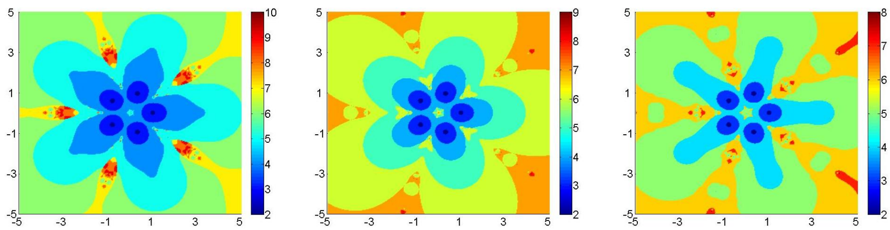
a	Time	MNI	CAI
$\sqrt{7}/2$	32.7823	4.098	1
$\sqrt{2}$	32.3302	4.0938	1
1.5	32.3851	3.9910	1
2	37.9665	4.1817	1
2.5	39.9706	4.3376	1
5	34.8024	4.1561	1

(b) method (16)

a	Time	MNI	CAI
$\sqrt{7}/2$	56.9791	7.4930	0.9726
$\sqrt{2}$	55.0062	7.3483	0.9808
1.5	54.3824	7.4701	0.9211
2	55.0616	7.3556	0.8732
2.5	63.5438	7.5176	0.8965
5	54.2124	7.1413	0.9674



(a) $a = \sqrt{3}/2$ (b) $a = \sqrt{7}/2$ (c) $a = 1.5$



(d) $a = 2$ (e) $a = 3$ (f) $a = 4$

Figure 5: Polynomiographs generated using (13), the convergence test (2) and $\alpha_n = f(a\sqrt{n})$, $\beta_n = g(a\sqrt{n^2+1})$ in the S-iteration

on f and g:

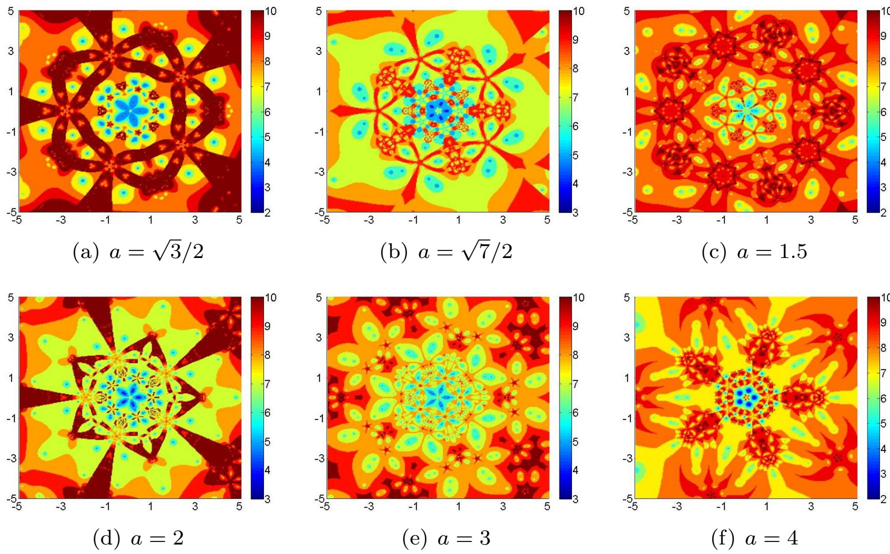
$$\alpha_n = f(a\sqrt{n}), \tag{21}$$

$$\beta_n = g(a\sqrt{n^2+1}), \tag{22}$$

where $a \in \mathbb{R}$.

The polynomiographs generated in $A = [-5, 5]^2$ using (13) and (16) are presented in Fig. 5 and 6, respectively. To generate these polynomiographs the standard convergence test (2) was used. The obtained polynomiographs show that the change of the a parameter in the α_n and β_n changes the shape of the polynomiographs in a significant way. Similar to the first example the shapes of obtained patterns are smooth and regular in the case of using (13), whereas in the case (16) the patterns are irregular and more intriguing. The finer details that are visible in polynomiographs obtained with (16) are more interesting in various artistic applications, e.g., as patterns on wallpapers or t-shirts, in creating paintings, carpet design etc.

The numerical results obtained during the generation of polynomiographs from Fig. 5 and 6 are presented in Tab. 3. From the results obtained with (13) we see that the generation time vary between 37 and 50 seconds and there is no obvious dependency of time on the parameter a. The times for (16) are longer (57 to 65 seconds), and also in this case there is no obvious dependency. The shortest time is obtained for $a = 4$ and $a = \sqrt{7}/2$ for (13) and (16), respectively. The difference of values of the MNI measure for both methods is significant. The MNI for (13) vary between 5.0877 (for $a = 4$) and 5.6567 (for $a = 3$), whereas



(a) $a=\sqrt{3}/2$ (b) $a=\sqrt{7}/2$ (c) $a=1.5$

(d) $a=2$ (e) $a=3$ (f) $a=4$

Figure 6: Polynomiographs generated using (16), the convergence test (2) and $\alpha_n = f(a\sqrt{n})$, $\beta_n = g(a\sqrt{n^2+1})$ in the S-iteration

for (16) between 7.5151 (for $\sqrt{7}/2$) and 8.5613 (for $a = \sqrt{3}/2$). Thus, method (13) converges faster than (16). When we look at the CAI measure we see that (13) obtains very high values of this measure, meaning that all or almost all points have converged to the roots. In the case of (16) we see that for all the considered values of a the CAI is less than 1, with the minimum equal to 0.6142 (for $a = \sqrt{3}/2$).

In the last example we present polynomiographs obtained with the same parameters as in the previous example, but with the convergence test given by (8). The polynomiographs for method (13) are presented in Fig. 7, whereas for (16) in Fig. 8. Comparing the polynomiographs from Fig. 5 (convergence

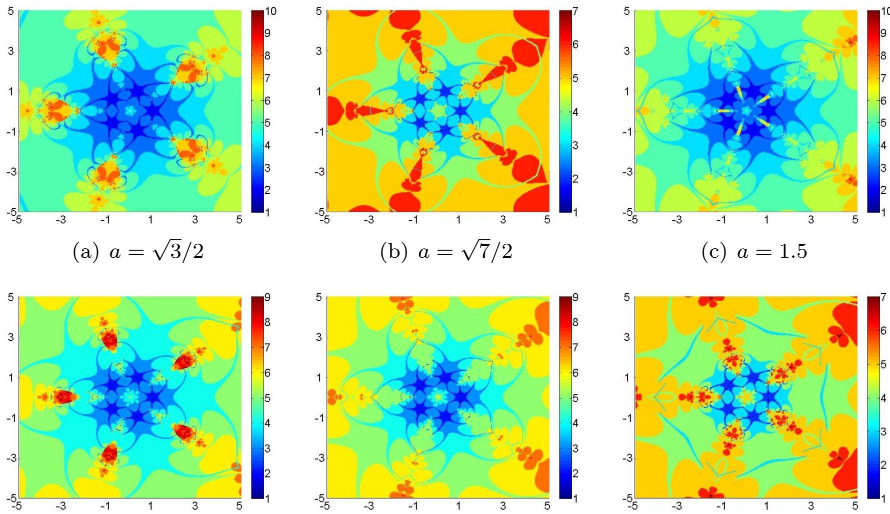
Table 3: Results obtained for the convergence test (2) and $\alpha_n = f(a\sqrt{n})$, $\beta = g(a\sqrt{n^2+1})$ in the S-iteration

(a) method (13)

a	Time	MNI	CAI
$\sqrt{3}/2$	40.6493	5.4401	0.9998
$\sqrt{7}/2$	40.4425	5.1911	1
1.5	43.1393	5.5042	0.9999
2	39.5907	5.3702	0.9999
3	50.3564	5.6567	1
4	37.9569	5.0877	1

(b) method (16)

a	Time	MNI	CAI
$\sqrt{3}/2$	65.1625	8.5613	0.6142
$\sqrt{7}/2$	57.2023	7.5151	0.9997
1.5	61.6361	8.3002	0.9351
2	63.2092	8.0267	0.8002
3	61.0711	8.0567	0.9230
4	60.0939	7.7808	0.9862



(a) $a = \sqrt{3}/2$ (b) $a = \sqrt{7}/2$ (c) $a = 1.5$

(d) $a = 2$ (e) $a = 3$ (f) $a = 4$

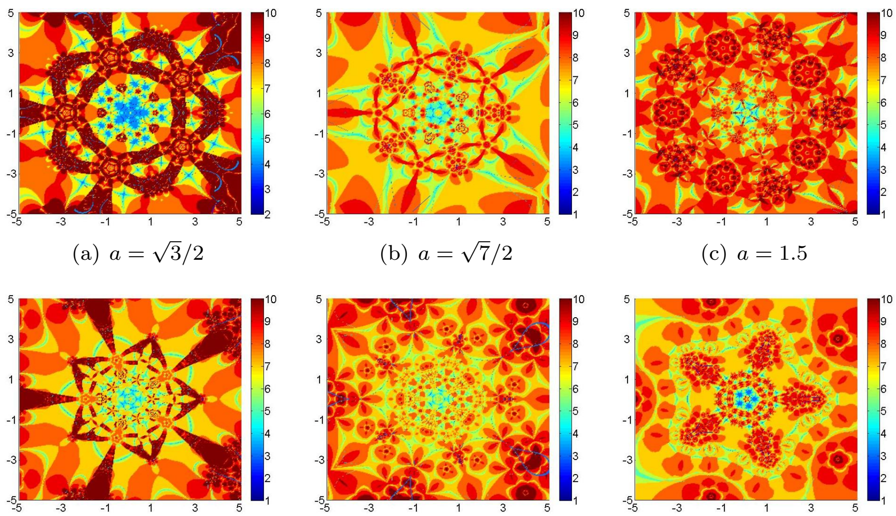
Figure 7: Polynomiographs generated using (13), the convergence test (8) and $\alpha_n = f(a\sqrt{n})$, $\beta_n = g(a\sqrt{n^2+1})$ in the S-iteration

test (2)) and 7 (convergence test (8)) we see that the use of the non-standard convergence test introduced interesting details into the polynomiograhs. Similar effect can be observed in the case of the use of (16) (compare polynomiographs from Fig. 6 and 8).

The numerical results obtained using the non-standard convergence test (8) and sequences of the parameters given by (21) nad (22) are gathered in Tab. 4. In both cases the times are shorter than in the corresponding cases for the standard convergence test presented in the previous example. The times are between 33 and 41 second for (13) and between 54 and 64 seconds for (16). In neither case we do not see any obvious dependency of time on the value of a. The lowest value of MNI for (13) is attained at $a = 4$. At the same time for $a = 4$ the method obtained the highest value of CAI, namely 1. So, each point in the considered area converged to a root. In the case of (16) the lowest value of MNI (7.2482) was attained at $a = \sqrt{7}/2$. And also in this case for the same value of a the method obtained the highest value of CAI, i.e., 0.9997. Thus, in the considered area one can find points that have not converged. Moreover, the lowest value of CAI for (16) is equal to 0.66, which shows a poor convergence of this method for $a = \sqrt{3}/2$.

5. Conclusions

In this paper, we presented some modifications of the root finding method presented in [10]. The first modification is modification of the formula. The second modification relays on the use instead of the standard Picard iteration of



(a) $a = \sqrt{3}/2$ (b) $a = \sqrt{7}/2$ (c) $a = 1.5$

(d) $a = 2$ (e) $a = 3$ (f) $a = 4$

Figure 8: Polynomiographs generated using (16), the convergence test (8) and $\alpha_n = f(a\sqrt{n})$, $\beta_n = g(a\sqrt{n^2+1})$ in the S-iteration

Table 4: Results obtained for the convergence test (8) and $\alpha_n = f(a\sqrt{n})$, $\beta = g(a\sqrt{n^2+1})$ in the S-iteration

(a) method (13)

a	Time	MNI	CAI
$\sqrt{3}/2$	34.0322	4.9294	0.9999
$\sqrt{7}/2$	34.3116	4.6960	1
1.5	36.4676	4.9439	0.9999
2	36.7840	4.8256	1
3	41.1271	5.1577	1
4	33.8420	4.5262	1

(b) method (16)

a	Time	MNI	CAI
$\sqrt{3}/2$	64.6580	8.3400	0.6600
$\sqrt{7}/2$	54.6322	7.2482	0.9997
1.5	63.6953	8.0557	0.9444
2	60.2835	7.9083	0.8241
3	59.1052	7.6932	0.9541
4	56.9820	7.5390	0.9908

the S-iteration with a periodic sequences of the parameters. Using the proposed methods and the periodic sequences we obtained polynomiographs of artistic value, especially in the case of the modified method. The numerical results show that the periodicity of the sequences used in the S-iteration has a significant effect on time, MNI and CAI. Using different periodicities we can obtain both better and worse values of the measures. The conducted experiments show that there is no obvious dependency of the measures on the periodicity of the sequences.

In future work, we will attempt to introduce the periodic parameters into other types of iteration process known in the literature. Moreover, we will numerically investigate the dependencies between the three measures, that were used in the paper, for the other iteration processes.

References

[1] B. Kalantari, Polynomiography and applications in art, education and science, Computers & Graphics 28 (3) (2004) 417–430. `doi:10.1016/j.cag.2004.03.009`.

[2] B. Kalantari, Polynomial Root-Finding and Polynomiography, World Scientific, Singapore, 2009. `doi:10.1142/9789812811837`.

[3] B. Kalantari, Two and three-dimensional art inspired by polynomiography, in: R. Sarhangi, R. Moody (Eds.), Renaissance Banff: Mathematics, Music, Art, Culture, Bridges Conference, Southwestern College, Winfield, Kansas, 2005, pp. 321–328.

[4] W. Kotarski, K. Gdawiec, A. Lisowska, Polynomiography via Ishikawa and Mann iterations, in: G. Bebis, R. Boyle, B. Parvin, D. Koracin, C. Fowlkes, S. Wang, M.-H. Choi, S. Mantler, J. Schulze, D. Acevedo, K. Mueller, M. Papka (Eds.), Advances in Visual Computing, Vol. 7431 of Lecture Notes in Computer Science, Springer, Berlin Heidelberg, 2012, pp. 305–313. `doi:10.1007/978-3-642-33179-4\_30`.

[5] S. Kang, H. Alsulami, A. Rafiq, A. Shahid, S-iteration scheme and polynomiography, Journal of Nonlinear Science and Applications 8 (5) (2015) 617–627.

[6] K. Gdawiec, W. Kotarski, A. Lisowska, Polynomiography based on the nonstandard Newton-like root finding methods, Abstract and Applied Analysis 2015 (2015) Article ID 797594, 19 pages. `doi:10.1155/2015/797594`.

[7] A. Rafiq, M. Tanveer, W. Nazeer, S. Kang, Polynomiography via modified Jungck, modified Jungck Mann and modified Jungck Ishikawa iteration scheme, PanAmerican Mathematical Journal 24 (4) (2014) 66–95.

[8] G. Ardelean, L. Balog, A qualitative study of Agarwal et al. iteration procedure for fixed points approximation, Creative Mathematics and Informatics 25 (2) (2016) 135–139.

[9] G. Ardelean, O. Cosma, L. Balog, A comparison of some fixed point iteration procedures by using the basins of attraction, Carpathian Journal of Mathematics 32 (3) (2016) 277–284.

[10] M. Bisheh-Niasar, A. Saadatmandi, Some novel Newton-type method for solving nonlinear equations, Boletim da Sociedade Paranaense de Matematica(in press).

[11] G. Alefeld, On the convergence of Halley's method, The American Mathematical Monthly 88 (7) (1981) 530–536. doi:10.2307/2321760.

[12] G. Ardelean, A comparison between iterative methods by using the basins of attraction, Applied Mathematics and Computation 218 (1) (2011) 88–95. doi:10.1016/j.amc.2011.05.055.

[13] C. Chun, B. Neta, S. Kim, On Jarratt's family of optimal fourth-order iterative methods and their dynamics, Fractals 22 (4) (2014) 1450013. doi:10.1142/S0218348X14500133.

[14] W. Mann, Mean value methods in iteration, Proceedings of the American Mathematical Society 4 (3) (1953) 506–510. doi:10.1090/S0002-9939-1953-0054846-3.

[15] S. Ishikawa, Fixed points by a new iteration method, Proceedings of the American Mathematical Society 44 (1) (1974) 147–150. doi:10.1090/S0002-9939-1974-0336469-5.

[16] R. Agarwal, D. O'Regan, D. Sahu, Iterative construction of fixed points of nearly asymptotically nonexpansive mappings, Journal of Nonlinear and Convex Analysis 8 (1) (2007) 61–79.

[17] M. Noor, New approximation schemes for general variational inequalities, Journal of Mathematical Analysis and Applications 251 (1) (2000) 217–229. doi:10.1006/jmaa.2000.7042.

[18] K. Gdawiec, W. Kotarski, Polynomiography for the polynomial infinity norm via Kalantaris formula and nonstandard iterations, Applied Mathematics and Computation 307 (2017) 17–30. doi:10.1016/j.amc.2017.02.038.

[19] K. Gdawiec, Polynomiography and various convergence tests, in: V. Skala (Ed.), WSCG 2013 Communication Papers Proceedings, Vaclav Skala – Union Agency, Plzen, Czech Republic, 2013, pp. 15–20.