Polynomiography and various convergence tests

Krzysztof Gdawiec
Institute of Computer Science
University of Silesia
Bedzinska 39
41-200, Sosnowiec, Poland
kgdawiec@ux2.math.us.edu.pl

ABSTRACT

The aim of this paper is to present a modification of the visualization process of finding the roots of a given complex polynomial which is called polynomiography. The name polynomiography was introduced by Kalantari. The polynomiographs are very interesting both from educational and artistic points of view. In this paper we are interested in the artistic values of the polynomiography. The proposed modification is based on the change of the usual convergence test used in the polynomiography, i.e. using the modulus of a difference between two successive elements obtained in an iteration process, with the tests based on distance and non-distance conditions. Presented examples show that using various convergence tests we are able to obtain very interesting and diverse patterns. We believe that the results of this paper can enrich the functionality of the existing polynomiography software.

Keywords

polynomiography, convergence, Basic Family, computer art

1 INTRODUCTION

One of the most elusive goals in computer aided design is artistic design and pattern generation. Pattern generation involves diverse aspects: analysis, creativity, development. A designer have to deal with all of these aspects in order to obtain an interesting pattern which later could be used in jewellery design, carpet design, as a texture etc. Therefore, it is highly motivating and useful to develop new methods of obtaining very diverse and interesting patterns. One place where we can search for this kind of methods is mathematics [Pic01].

Polynomials are one of the mathematical objects which can generate very diverse and beautiful patterns. The patterns from polynomials are often generated through polynomiography. It visualizes the process of finding roots of a complex polynomial using the numerical methods. In this paper we are not interested in the improvement of the numerical methods convergence, but in the artistic aspect of the polynomiography. This aspect includes: creating paintings, carpet design, tapestry design, animations etc. [Kal05b]. So we are interested in obtaining new and interesting patterns basing on the theory of polynomiography.

The paper is organized as follows. In section 2 we introduce the basics of polynomiography. At first we define the Basic Family and give an efficient algorithm for computation of a value for a given element of this family and an algorithm for computation of polynomiograph. The section ends with some examples of polynomiographs. Next, in section 3 we introduce different kinds of convergence test which can be used in the algorithm of polynomiograph computation. In section 4 we show some examples of polynomiographs obtained using the proposed convergence tests. Finally, in section 5 we give concluding remarks and plans for the future work.

2 POLYNOMIOGRAPHY

Polynomiography was introduced by Kalantari about 2000. It is "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" [Kal04]. Single image created using the mentioned methods is called polynomiograph. In 2005 Kalantari obtained an U.S. patent on the use of polynomiography in the generation of aesthetic patterns [Kal05a].

In mathematics polynomials are fundamental objects with very diverse applications, e.g. in error correcting codes, interpolation, engineering etc. From the Fundamental Theorem of Algebra we know that a polynomial of degree n with complex coefficients has n roots which may or may not be distinct. The problem of finding the roots of a given polynomial was known since the Sumerians, i.e. 3000 BC. Since then many different methods of finding the roots approximation were proposed, e.g. Newton's method [Var02], Harmonic Mean Newton's method [Ard11], Whittaker's method [Var02], Halley's method [Ard11], Chebyshev's method [Var02], Traub-Ostrowski's method [Var02] etc.

Let us consider a polynomial $p \in \mathbb{C}[Z]$ and $\deg p \geq 2$ of the form:

$$p(z) = a_n z^n + a_{n-1} z^{n-1} + \ldots + a_1 z + a_0. \qquad (1)$$

Now we define a sequence of functions $D_m : \mathbb{C} \to \mathbb{C}$ for all $z \in \mathbb{C}$ [Kal09]:

$$D_0(z) = 1,$$

$$D_m(z) = \det \begin{pmatrix} p'(z) & \frac{p''(z)}{2!} & \cdots & \frac{p^{(m-1)}(z)}{(m-1)!} & \frac{p^{(m)}(z)}{m!} \\ p(z) & p'(z) & \ddots & \ddots & \frac{p^{(m-1)}(z)}{(m-1)!} \\ 0 & p(z) & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \frac{p''(z)}{2!} \\ 0 & 0 & \cdots & p(z) & p'(z) \end{pmatrix} \qquad (2)$$

for $m \geq 1$.

Using the D_m sequence we define a Basic Family $\{B_m\}_{m=2}^{\infty}$, where $B_m : \mathbb{C} \to \mathbb{C}$, in a following way [Kal09]:

$$\forall_{z \in \mathbb{C}} \quad B_m(z) = z - p(z) \frac{D_{m-2}(z)}{D_{m-1}(z)}. \qquad (3)$$

The Basic Family is a fundamental part of polynomiography. Let us see how the first three elements of the Basic Family look like:

$$B_2(z) = z - \frac{p(z)}{p'(z)}, \qquad (4)$$

$$B_3(z) = z - \frac{2p'(z)p(z)}{2p'(z)^2 - p''(z)p(z)}, \qquad (5)$$

$$B_4(z) = z - \frac{6p'(z)^2 p(z) - 3p''(z)p(z)^2}{p'''(z)p(z)^2 + 6p'(z)^3 - 6p''(z)p'(z)p(z)}. \qquad (6)$$

As we look at those formulas we see that B_2 is formula used in Newton's root finding method, and B_3 is formula used in Halley's method. Moreover, we see that when m increases the formula for B_m becomes more and more complex. So we need an efficient algorithm for its computation. In [Kal10] Kalantari introduced such algorithm (Algorithm 1). To derive this algorithm he used the theory of symmetric functions.

Algorithm 2 presents a method of determining polynomiograph [Kal09]. In the algorithm for each point in the considered area $A \subset \mathbb{C}$ we iterate given element of the Basic Family (defined by $p \in \mathbb{C}[Z]$ and $m \geq 2$). If

Algorithm 1: $B_m(z)$ computation

Input: $p \in \mathbb{C}[Z]$, $\deg p \geq 2$ – polynomial, $m \geq 2$ – number for B_m, $z_0 \in \mathbb{C}$ – point for which we make the computations.
Output: $B_m(z_0)$.

```
h[0] = 1
for i = 0 to m − 1 do
    e[i] = p^(i)(z0)/(i! p(z0))
for i = 1 to m − 1 do
    h[i] = Σ_{r=0}^{i−1} (−1)^{i−r−1} e[i − r] h[r]
B_m(z0) = z0 − h[m − 2]/h[m − 1]
```

the modulus of the difference between two successive points in the iteration process is smaller than the given accuracy $\varepsilon > 0$ we assume that the generated sequence converge to a root of p and we stop iterating. If we reach the maximum number of iterations k we assume that the generated sequence do not converge to any root of p. At the end we give a colour to the considered point using the given colourmap and the iteration number at which we have left the while loop.

Algorithm 2: Polynomiograph computation

Input: $p \in \mathbb{C}[Z]$, $\deg p \geq 2$ – polynomial, $A \subset \mathbb{C}$ – area, k – number of iterations, ε – accuracy, $m \geq 2$ – number for B_m, $colours[0..k]$ – colourmap.
Output: Polynomiograph for the area A.

```
for z0 ∈ A do
    i = 0
    while i ≤ k do
        z_{i+1} = B_m(z_i)
        if |z_{i+1} − z_i| < ε then
            break
        i = i + 1
    Print z0 with colours[i] colour
```

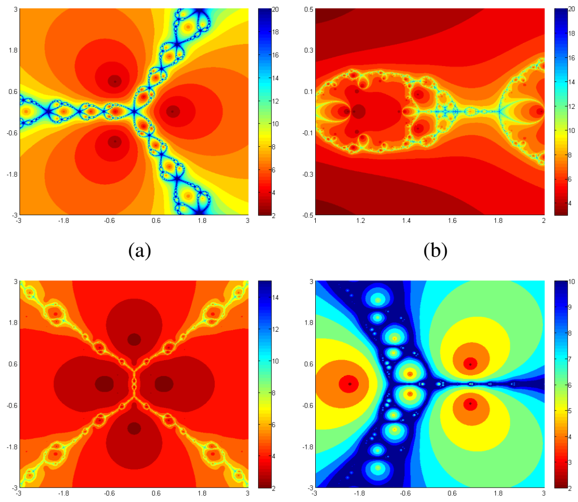
Examples of polynomiographs generated using Algorithm 2 for:

(a) $p(z) = z^3 - 1$, $A = [-3,3]^2$, $k = 20$, $\varepsilon = 0.001$, $m = 2$,

(b) $p(z) = -2z^4 + z^3 + z^2 - 2z - 1$, $A = [1,2] \times [-0.5, 0.5]$, $k = 20$, $\varepsilon = 0.001$, $m = 3$,

(c) $p(z) = z^4 + z^2 - 1$, $A = [-3,3]^2$, $k = 20$, $\varepsilon = 0.001$, $m = 4$,

(d) $p(z) = z^3 - 3z + 3$, $A = [-3,3]^2$, $k = 10$, $\varepsilon = 0.001$, $m = 2$



(a) (b) (c) (d)

Figure 1: Examples of polynomiographs.

are presented in Fig. 1.

In Algorithm 2 to colour the points we use the iteration number for which we have left the iteration process, we call this method the iteration colouring. We can use different methods of colouring, e.g. basins of attraction (each polynomial root has its own colour, for each point in A we iterate it and when the condition in line 5 of Algorithm 2 is meet the considered point gets the colour of the nearest root), mixed method (we mix the iteration colouring and the basins of attraction) etc. [Kal09].

3 DIFFERENT CONVERGENCE TESTS

In line 5 of Algorithm 2 we see a standard test for convergence of an iteration process in the numerical root finding methods. In the test we take two elements: the one computed in the current iteration and the element from the previous iteration, and we calculate the modulus of their difference. Then we check if the calculated value is smaller than the given accuracy. The modulus calculation in the test is equivalent to the computation of the distance between these two points of the complex plane. So we may change the way in which we calculate the distance with a different metric.

We know that the complex plane $\mathbb{C}$ is isometric with $\mathbb{R}^2$, where the isometry $\phi : \mathbb{C} \to \mathbb{R}^2$ is defined as follows [Sea07]:

$$\phi(z) = (\Re(z), \Im(z)) \tag{7}$$

for every $z \in \mathbb{C}$, and where $\Re(z)$, $\Im(z)$ denote the real and imaginary part of z (respectively). Using the isometry we can define metric $d : \mathbb{C} \times \mathbb{C} \to [0, +\infty)$ using metric $\rho : \mathbb{R}^2 \times \mathbb{R}^2 \to [0, +\infty)$ in a following way [Sea07]:

$$d(z_1, z_2) = \rho(\phi(z_1), \phi(z_2)), \tag{8}$$

where $z_1, z_2 \in \mathbb{C}$.

On $\mathbb{R}^2$ we have many different metrics which we may use [Sea07], e.g.

- taxicab metric
$$\rho((x_1, y_1), (x_2, y_2)) = |x_1 - x_2| + |y_1 - y_2|, \tag{9}$$
- supremum metric
$$\rho((x_1, y_1), (x_2, y_2)) = \max\{|x_1 - x_2|, |y_1 - y_2|\}, \tag{10}$$
- l_p metric
$$\rho((x_1, y_1), (x_2, y_2)) = [|x_1 - x_2|^p + |y_1 - y_2|^p]^{\frac{1}{p}}, \tag{11}$$
where $1 \leq p \leq +\infty$.

When we have some metric space (X, ρ) we can define new metrics using following facts [Sea07]:

- if $f : X \to X$ is injective, then
$$\eta(x, y) = \rho(f(x), f(y)) \tag{12}$$
is a metric on X,
- if $f : X \to \mathbb{R}$ is a function, then
$$\eta(x, y) = \rho(x, y) + |f(x) - f(y)| \tag{13}$$
is a metric on X.

From the examples presented in the next section we will see that changing the metric produces only a small change in the shape of polynomiograph. As we are interested in generation of interesting patterns using the polynomiography and not in the best convergence of the numerical method we can relax the assumption about the metric. For this purpose we can take $p \in (0, 1)$ in the l_p metric obtaining the so-called fractional distance which is used for instance in models for forecasting pollution concentrations [DW12].

We also can omit the assumption about the injectivity of f in (12). For instance when we take $\mathbb{C}$ with the modulus metric and $f(z) = |z|^2$, which is not injective, we obtain:

$$\eta(z_1, z_2) = ||z_1|^2 - |z_2|^2|. \tag{14}$$

The η function from (14) was used instead the modulus test by Pickover in Halley's method in [Pic88]. In this way Pickover obtained very diverse shapes of the polynomiographs.

Another way to modify the tests is to add some weights in the metric functions. When we use (12) we can add two weights $\alpha, \beta \in \mathbb{R}$ in a following way:

$$\eta(x, y) = \rho(\alpha f(x), \beta f(y)). \tag{15}$$

In this way we loose the metric property of η, e.g. it is not symmetric for $\alpha \neq \beta$, but as we will see in section 4 we obtain very diverse polynomiographs using this function.

Till now the proposed tests were based on metrics, but there is no obstacle in using tests which are based on functions that are not metric, quasimetrics etc. at all. For instance we can use following tests:

$$|\exp(\alpha z_{i+1} - \beta z_i)| < \varepsilon, \tag{16}$$

$$|\alpha \Re(z_{i+1} - z_i)| < \varepsilon \vee |\beta \Im(z_{i+1} - z_i)| < \varepsilon, \tag{17}$$

$$|\alpha \Re(z_{i+1} - z_i)|^2 < \varepsilon \wedge |\beta \Im(z_{i+1} - z_i)|^2 < \varepsilon, \tag{18}$$

where $\alpha, \beta \in \mathbb{R}$. In the tests which consist of several terms joined with logical operators, e.g. (17), (18), instead of one ε we can use separate value for each term.

The last group of tests which we propose is based on the idea taken from the escape time algorithm which is used in the Julia set drawing. Similar like in the escape time algorithm we can check if a value of some iterated function escapes, i.e. is greater than the given radius $R > 0$. Examples of this kind of tests are:

$$|z_{i+1} - z_i| + |\arg(z_{i+1}) - \arg(z_i)| > R, \tag{19}$$

$$\left| \frac{1}{|z_{i+1}|^2} - \frac{1}{|z_i|^2} \right| + ||z_{i+1}|^2 - |z_i|^2| > R, \tag{20}$$

$$\alpha |\Re(z_{i+1} - z_i)| > R \wedge \beta |\Im(z_{i+1} - z_i)| > R, \tag{21}$$

where $\arg(z)$ is an argument of complex number z, and $\alpha, \beta \in \mathbb{R}$.

4 EXAMPLES

In this section we show some examples of using the different tests proposed in section 3. We start our examples with changing the standard metric (modulus) used in the polynomiography with the supremum metric. In the example we use: $p(z) = z^3 - 3z + 3$, $A = [-2, 2]^2$, $k = 15$, $\varepsilon = 0.001$, $m = 2$. Figure 2(a) presents the result for the modulus metric and Fig. 2(c) presents the result for the supremum metric. From the figures we see that in both cases the result is very similar and the difference is small. To see the difference more precisely in Figs. 2(b), 2(d) magnification of the marked areas from Figs. 2(a), 2(c) are presented. In the case of modulus metric we have smooth boundaries between the regions and for the supremum metric the boundaries are frayed and the regions are lighter. When we use a different metric instead of the supremum metric the effect will be very similar, so the obtained results are not interesting from the artistic point of view.

In the next example we use the test used by Pickover (14) and its weighted modification. The common parameters used in the example: $p(z) = z^4 + z^2 - 1$, $A = [-3, 3]^2$, $k = 15$, $\varepsilon = 0.001$, $m = 2$. Figure 3(a) presents

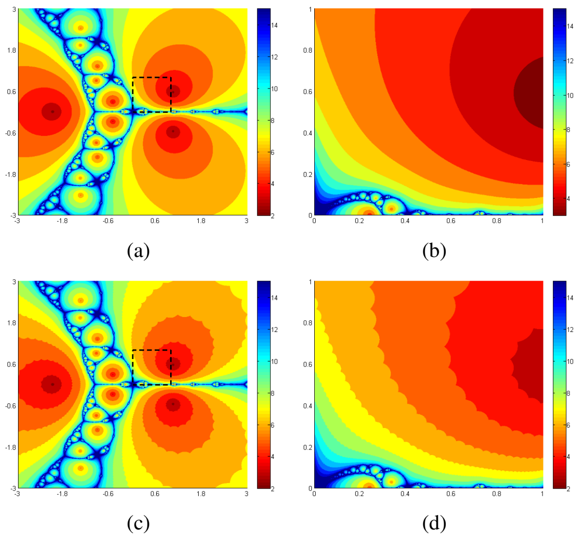


Figure 2: Examples of polynomiographs: (a) with modulus metric, (b) with supremum metric, (c) magnification of the marked area from (a), (d) magnification of the marked area from (c).

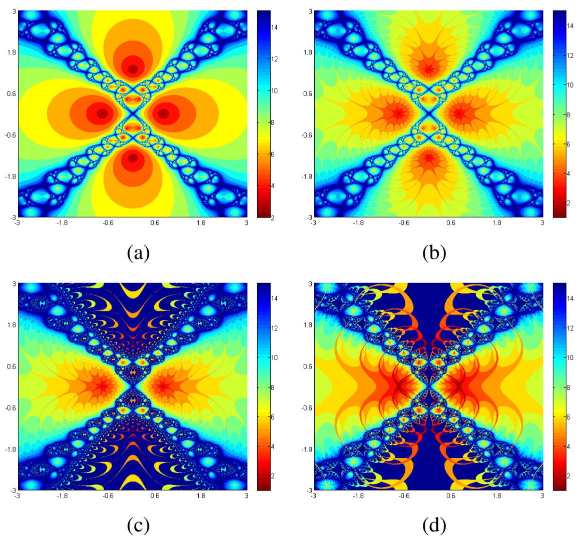


Figure 3: Examples of polynomiographs: (a) original, (b) using the Pickover test, (c), (d) using the weighted version of Pickover test.

the result for the original test, Fig. 3(b) for the Pickover test and Figs. 3(c), 3(d) the results for weighted version of (14), i.e. $|\alpha|z_1|^2 - \beta|z_2|^2|$, where $\alpha = 1.05$, $\beta = 1.049$ for (c) and $\alpha = 0.049$, $\beta = 0.05$ for (d).

The Pickover test changes the regions of polynomiograph where the convergence using the original test was fast. In this way we obtain some swirls in the smooth areas. Using the test with weights we obtain even more changes in the areas of the fast convergence and moreover small changes in the areas of the slow convergence. The polynomiographs obtained with the non-standard test look very interesting and the patterns are more complex comparing to the original one.

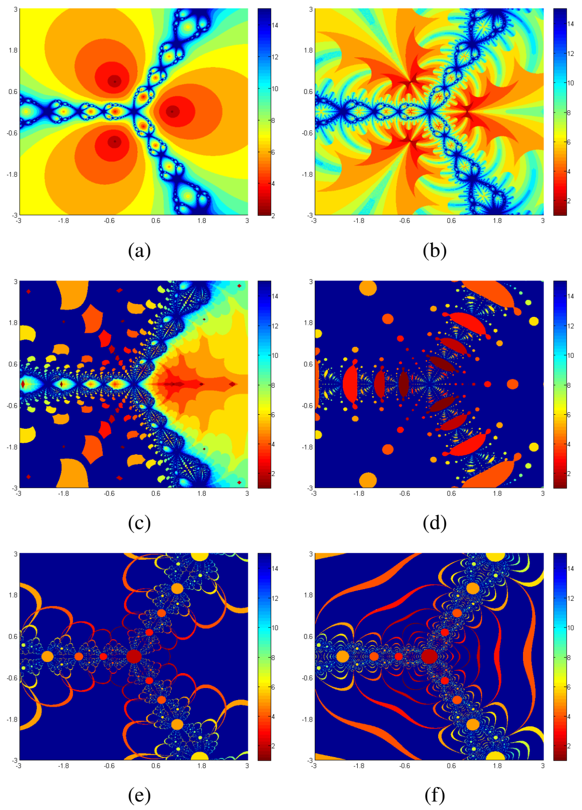


Figure 4: Examples of polynomiographs with different tests based on metrics and weights.

In the previous example we used only the Pickover test and now we show examples of more tests which are based on metrics and weights. The common parameters used in the example: $p(z) = z^3 - 1$, $A = [-3,3]^2$, $k = 15$, $\varepsilon = 0.001$, $m = 2$. Figure 4(a) presents the original polynomiograph and Figs. 4(b)-(f) present polynomiographs obtained with the help of different metrics and weights. The tests used in the example were following:

(a) $|z_{i+1} - z_i| < \varepsilon$,

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

(c) $|0.05\sin(\Re(z_{i+1})) - 0.049\sin(\Re(z_i))| + |0.05\sin(\Im(z_{i+1}) - 0.049\sin(\Im(z_i))| < \varepsilon$,

(d) $|0.01z_{i+1}^{10} - 0.008z_i^{10}| < \varepsilon$,

(e) $|\frac{0.05}{|z_{i+1}|^2} - \frac{0.045}{|z_i|^2}| < \varepsilon$,

(f) $|\frac{0.045}{|z_{i+1}|^2} - \frac{0.05}{|z_i|^2}| < \varepsilon$.

From the presented polynomiographs we see that using the different metrics and weights we are able to obtain very diverse and interesting patterns comparing to the original test. In the Fig. 4(b) we can observe a pattern which looks like a knot and in Fig. 4(e) pattern which reminds a flower. From Fig. 4(e) and Fig. 4(f) we see that the patterns look quite different, but the tests used for their creation differ only in order of the weights (they are interchanged).

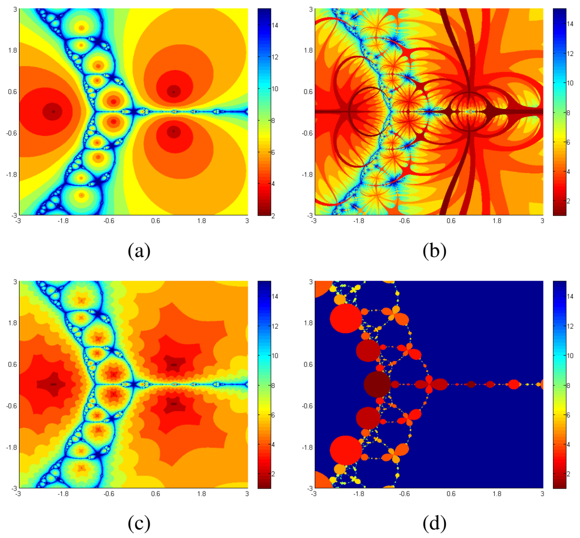


Figure 5: Examples of polynomiographs: (a) original, (b)-(d) based on the non-metric tests.

Next example presents the use of the tests which are based on the non-metric conditions. The common parameters used in the example: $p(z) = z^3 - 3z + 3$, $A = [-3,3]^2$, $k = 15$, $\varepsilon = 0.001$, $m = 2$. Figure 5(a) presents the original polynomiograph and Figs. 5(b)-(d) present polynomiographs obtained with the help of following tests:

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

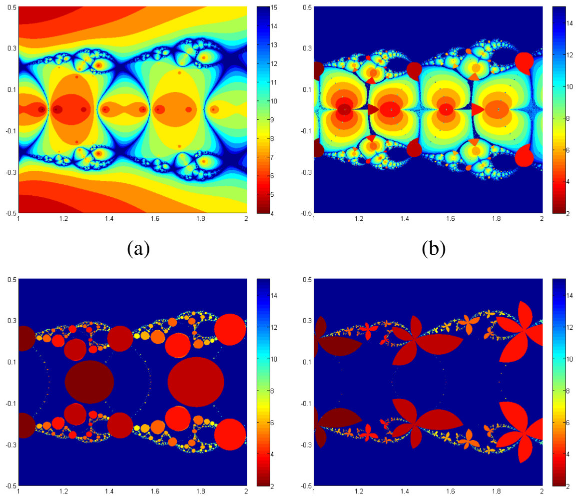
(c) $|0.4\Re(z_{i+1} - z_i)|^2 < \varepsilon \wedge |\Im(z_{i+1} - z_i)|^2 < \varepsilon$,

(d) $|\exp(10z_{i+1} - 9z_i)| < \varepsilon$.

Also in this case we see that when we change the modulus test to the tests based on the non-metric conditions we obtain very interesting patterns. For instance in Fig. 5(b) we see a very complicated net of swirls and in Fig. 5(d) a pattern which looks like a necklace.

In the last example we show some polynomiographs obtained with the tests basing on the escape criteria. The common parameters used in the example: $p(z) = -2z^4 + z^3 + z^2 - 2z - 1$, $A = [1,2] \times [-0.5, 0.5]$, $k = 15$, $m = 2$. Figure 6 presents the original polynomiograph for $\varepsilon = 0.001$ and Figs. 6(b)-(d) present polynomiographs obtained with the help of following tests: (b) $R = 6$ and condition (19), (c) $R = 8$ and condition (20), (d) $R = 6$ and condition (21) for $\alpha = 8$ and $\beta = 11$.

The patterns obtained with the escape criteria also differ from the original one. But obtaining a very interesting pattern using those criteria is difficult. This is because the patterns arise in the regions where the original method converges very slowly or reaches the maximum number of iterations.



(a) (b) (c) (d)

Figure 6: Examples of polynomiographs: (a) original, (b)-(d) based on the escape criteria.

5 CONCLUSIONS

In this paper we presented modifications of the polynomiography algorithm. The modifications were based on the change of the usual convergence test with the tests based on distance and non-distance conditions. Presented examples show that using the proposed tests we are able to obtain very interesting patterns. We believe that the results of this paper can enrich the functionality of the existing polynomiography software.

When we search for an interesting pattern using the polynomiography we must make the right choice of a polynomial, the iteration function etc. and using the trial and error we must find an interesting area [Kal09]. Adding our tests to the list of polynomiography parameters we make the search even more difficult, so there is a need for automatic method which finds interesting patterns. The notion of an interesting pattern is very difficult to define and usually is based on a subjective feeling, but there are some attempts to estimate the notion. Ashlock and Jamieson in [AJ08] introduced a method of exploring the Mandelbrot and Julia sets for interesting patterns. They used evolutionary algorithms with different fitness functions. In our further research we will try to develop a method which searches for interesting patterns in the polyniomography using similar methodology like that presented by Ashlock.

Polynomiography is based on the complex polynomials. In [Lev94] we can find examples of using q-systems numbers instead of complex numbers for obtaining diverse patterns, and in [WS13] we find bicomplex numbers used in the Mandelbrot and Julia sets. Using the q-system and bicomplex numbers in the polynomiography can probably further enrich the patterns obtained with the polynomiography.

6 REFERENCES

[Ard11] Ardelean, G.: A Comparison Between Iterative Methods by Using the Basins of Attraction. Applied Mathematics and Computation 218(1), 88-95, (2011)

[AJ08] Ashlock, D., Jamieson, B.: Evolutionary Exploration of Complex Fractals. In: P.F.Hingston, L.C.Barone, Z.Michalewicz (eds.) Design by Evolution. Springer, Berlin, pp. 121-143, (2008)

[DW12] Domańska, D., Wojtylak, M.: Application of Fuzzy Time Series Models for Forecasting Pollution Concentrations. Expert Systems with Applications 39(9), 7693-7679, (2012)

[Kal04] Kalantari, B.: Polynomiography and Applications in Art, Education and Science. Computers & Graphics 28(3), 417-430, (2004)

[Kal05a] Kalantari, B.: Method of Creating Graphical Works Based on Polynomials. U.S. Patent 6,894,705, issued May 17, 2005

[Kal05b] Kalantari, B.: Two and Three-dimensional Art Inspired by Polynomiography. Proceedings of Bridges, Banff, Canada, pp. 321-328, (2005)

[Kal09] Kalantari, B.: Polynomial Root-Finding and Polynomiography. World Scientific, Singapore (2009)

[Kal10] Kalantari, B.: A Combinatorial Construction of High Order Algorithms for Finding Polynomial Roots of Known Multiplicity. Proceedings of the American Mathematical Society 138(6), 1897-1906, (2010)

[Lev94] Levin, M.: Discontinuous and Alternate Q-System Fractals. Computer & Graphics 18(6), 873-884, (1994)

[Pic88] Pickover, C.A.: A Note on Chaos and Halley's Method. Communications of the ACM 31(11), 1326-1329, (1988)

[Pic01] Pickover, C.A.: Computers, Pattern, Chaos, and Beauty: Graphics from an Unseen World. Dover Publications, Mineola, (2001)

[Sea07] Searcóid, M.Ó: Metric Spaces. Springer, London, (2007)

[Var02] Varona, J.L.: Graphics and Numerical Comparison Between Iterative Methods. The Mathematical Intelligencer 24(1), 37-46, (2002)

[WS13] Wang, X.-Y., Song, W.-J.: The Generalized M-J Sets for Bicomplex Numbers. Nonlinear Dynamics 72(1-2), 17-26, (2013)