

Nonlinear Space Transformations and Educational Software: Discrete-time dynamical systems¹

Vladimir Nodelman
nodelman@hit.ac.il

Department of Computer Sciences,
Holon Institute of Technology
52 Golomb St., Holon, 58102,
ISRAEL

Abstract

*The possibility to study such fundamental notions of modern mathematics as “space” and “spatial transformation” is almost absent in educational software. Instead, this software handles **affine transformations of objects** lying in 2D or 3D space: specifically, compositions of rotations, translations and homothety, all applied to geometric figure.*

*It is relatively easy to implement affine transformations of the **whole space** programmatically due to internal nature of computer graphics mechanism. The problem is to support **nonlinear spatial transformations** in a manner that is user-friendly and seamless.*

*This is the last part of three-paper series that presents the author’s noncommercial software “VisuMatica”, its **2D- and 3D-nonlinear transformational abilities and their didactic potential**. As result, the software becomes a powerful tool, which helps to discover the unity of mathematics, to visualize and dynamically explore new mathematical environments and phenomena. In particular, the paper includes discussion of nonlinear space transformations’ application to studies of algebra, complex analysis, vector fields, differential equations and modeling.*

1. Introduction

In the opening article of this three-paper series, about nonlinear spatial transformations (NLST), we discussed their role in mathematics and the specifics of NLST support by educational software. *Not formally*, we classified NLST supported by *VisuMatica* by the *transforming action* (**one-step**, **continuous**, and **discrete**). The three papers examine the specifics of educational software support of studies of these transforming actions separately. This paper is the last part in the series.

The students’ mental activities while studying NLST, mainly, are the same as in case of any other transformations. NLST, being functions, require consolidation of the following two complementary actions:

- *Finding image of an element or a subset of the Domain,*
- *Finding preimage of an element or a subset of the Range,*

which acquire a special meaning in the context of NLST, discussed in this paper.

Although these activities are the most important and their assimilation includes inter alia mastering mapping, educational software has to support deeper and wider exploration of specific and common features and behavior of the studied NLST.

The following case studies present the *VisuMatica*’s support of these pedagogical tasks. We selected these examples to emphasize the strength of educational software as a tool for deep and

¹See *colored figures and animations*: <https://sites.google.com/site/nonlinearspacetransformations/>

wide exploration of the studied subject that allows students to discover new and surprising features “everywhere”, even in a “boring” and/or well-known math.

2. Discrete-time dynamical systems

Transition from continuous-time orbit $\varphi(x, t)$ to discrete-time sequence $\varphi(x, n)$, where n as time takes integral values starting from the initial condition at $t = 0$, is a common way in making them suitable for numerical evaluation on computer. The technique of continuous-time step-by-step modelling, considered in the previous paper, does it explicitly. The so smooth looking flood-filling animations implicitly utilize the same practice as well.

2.1. 1D mapping

We will consider here the *recurrence relation* of form $x_{n+1} = f(x_n)$. By setting the initial term x_0 , each subsequent term is determined by this relation. This interpretation of $f(x)$ as a recurrent mapping of its arguments opens a new, fresh view on the notion of function. Comprehensible even for high school students, this meaning provides a rich educational basis to widen their understanding of functional mapping. It looks simple and easy, but, in fact, allows exploring very deep and not trivial mathematics.

The necessary software support of these explorations we will illustrate by considering a classical example of nonlinear discrete-time dynamical system - the logistic map $x_{n+1} = ax_n(1 - x_n)$. By varying the initial term x_0 and value of parameter a we will observe the correspondent series-orbits $x(x_0, n)$. As before, our attention will be concentrated on fixed points, cycles and, in addition, on chaotic behavior.

Fig.1 presents graph of the right-side function $y = ax(1 - x)$ with addition of so called *cobweb plot* or *stair-step diagram*. Here $a = 2$, $x, y \in [0,1]$. The first and third images match to $x_0 = 0.015$ and the second one – to $x_0 = 0.756$.

Fig.1 presents also the user interface of featuring the stair-step diagram. One can define the value of x_0 manually, by slider of scrollbar or set its exact value or a proper expression straight out in the edit box. The depth of recursion may be changed as well. The n^{th} step of diagram colored proportionally to the palette, with ratio n/depth .

This rather simple environment helps to discover many features of the mapping. First of all, the following fact:

The initial value of x_0 does not imply to the tendency. In case of $a = 2$ all orbits converge to the *fixed point* 0.5.

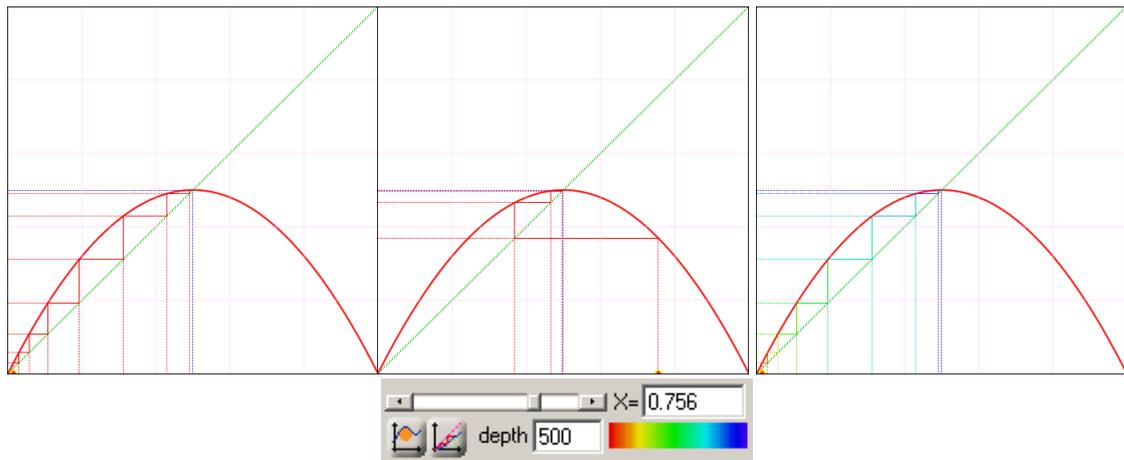


Figure 1

The stairs-step's coloring scheme of *VisuMatica* allows emphasize the qualitative character of series behavior. Thus, all the few stair steps in the first two images (depth equals to 500) colored in the tints on red – the starting color in the palette. The only different color fits the fixed value of 0.5, and it is blue. Increasing the depth value does not change the image visibly – neither colors, nor steps! The third image in Fig.1 was received by changing the depth value to 7. We see here the same trace as in the first one ($x_0 = 0.015$) but the multicolored stairs currently are presenting the seven equally distributed tints of the palette. Students now can easily explain the all-red – one-blue effect as result of overdrawing of, say 500 – 6, last steps by the latest tint of the palette, the blue one. Remains to conclude that:

x_n series converges to 0.5 very quickly. The majority of the steps sit very close to the fixed point.

VisuMatica adds an educational touch to the quantitative portray of cobweb plot by projecting the steps onto coordinate axes. This way it visually refers to the mechanism of recurrent construction. - The marching squares with common vertex at the origin emphasize the sequence: *each calculated value of the function becomes an argument of the next step of evolution.*

In order to provide deeper details in visualization we can use *time series show* in an additional small view. Points with coordinates (n, x_n) present here the discrete-time sequence. Fig.2 illustrates different cases of sequence behavior by means of synchronized stair-step diagram and time series show as follows²:

² To animate the value of x_0 we assign $x_0 = b$, define parameter b in interval $[0,1]$, and change its value by slider or by the “animation” button .

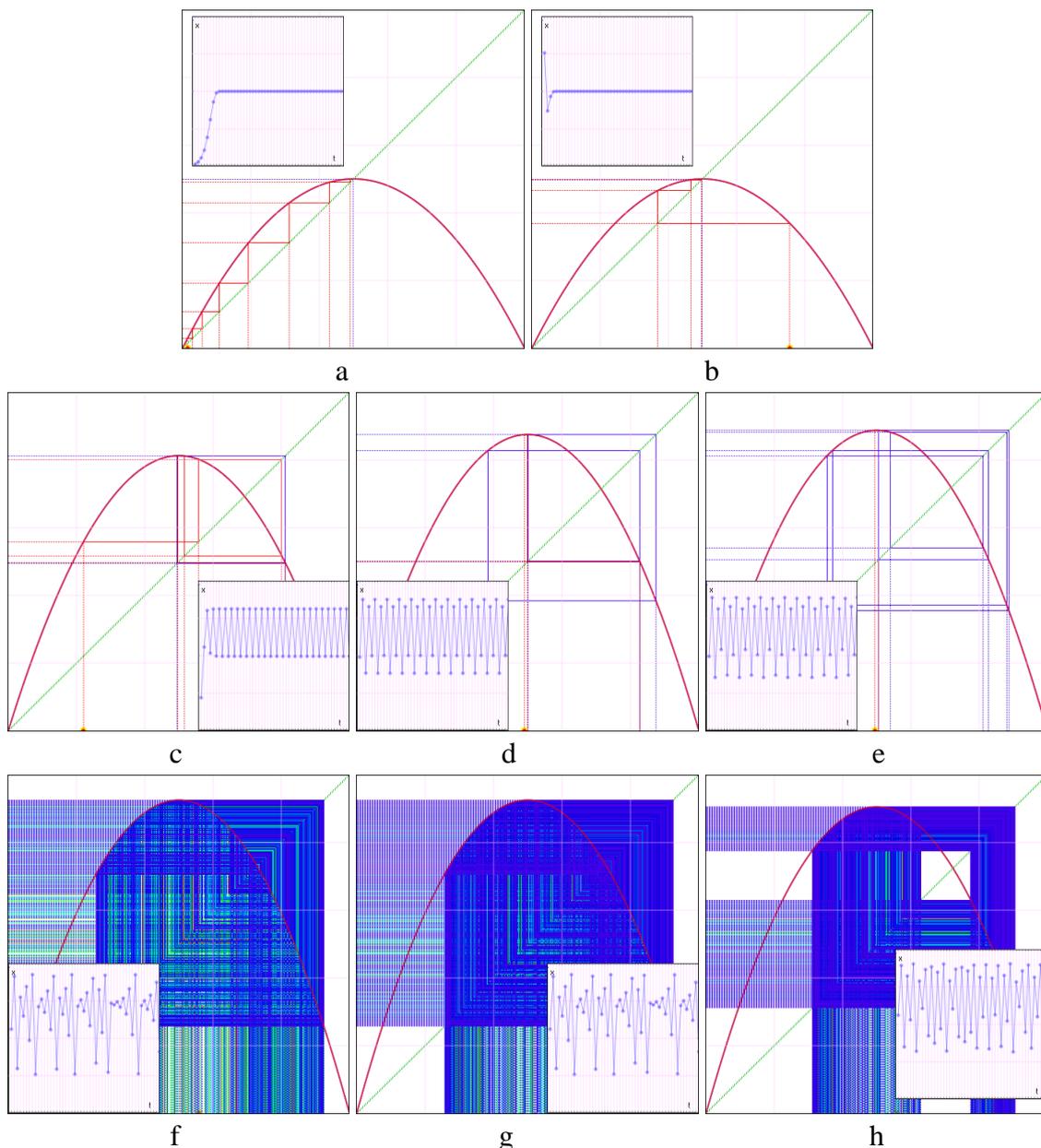


Figure 2

- Case a), b) are the same as before. The quick convergence to the fixed point is clearly visible in the small view.
-
- Case c), where $a = 3.25$ and $x_0 = 0.22$ demonstrates the cyclic behavior between two approximate values: 0.49 and 0.81 (period 2).
-
- Case d), where $a=3.5$ and $x_0 = 0.49$ shows cyclic behavior with already four values 0.38, 0.83, 0.50, and 0.87 (period doubles: 4).
-
- Case e), where $a=3.55$ and $x_0 = 0.495$ shows cyclic behavior with eight values 0.89, 0.35, 0.81, 0.54, 0.88, 0.37, 0.83 and 0.50 (period doubles: 8).

- Cases f) and g), where $a=3.7$ and $x_0 = 0.56$ and h) with $a=3.62$ and $x_0 = 0.56$ show *chaotic* behavior. The term “*chaotic*” is not occasional. We can see the “crazy” behavior with different depth values (500 in f) and 5000 in g)). And the chaotic “craziness” expressed here in few ways:
 - The 50 starting elements of the sequence-orbit in *time series* are clearly the same. Already here, in contrast to cases a)-d), it is impossible to recognize any regularity.
 - The blue color (the ending tint of the palette) *randomly* distributed over *the segment/segments of x-axis* covered by x_n . The non-blue gaps there have different and randomly distributed colors.
 - It seems that by increasing depth of recursion one can “*completely*” cover the segment/segments.
 - In addition to the noted fact of independency of the orbit’s behavior from the initial value of x_0 , one can also discover independency of the endpoints of the segment from x_0 .

The next step in exploration of discrete-time dynamic system naturally assumes an analysis of its actual dependence on the value of a , which till now seems very strange. Small changes of the parameter a value caused sudden qualitative changes in its behavior. Such phenomenon – so called *bifurcation* - has a visual expression by *bifurcation diagrams*. A *bifurcation diagram* shows the possible sudden behavior of a qualitatively different solutions (fixed point, periodic orbit, chaos) as a function of a *bifurcation parameter* (a in our case of logistic map).

VisuMatica portrays bifurcation diagrams in form of full or small view³. The left image in Fig.3 presents the bifurcation diagram in a small view together with stair-step diagram (the values are the same as in Fig.2 h).

The abscissa axis here presents $a \in [0, 4]$, and the ordinate axis presents values x_n of the correspondent orbit-sequence. The yellow-red dotted vertical line emphasizes the current value of a (3.62). A close look on the points of diagram, which lie on the red line⁴, allows recognize the fact that we have already discovered: if $a = 3.62$ the solution behaves chaotically inside two separate segments. In Fig.3 b) we brought together the red lines, which are fitting different orbits from Fig.2. It confirms our conclusions about behavior and features of these orbits.

Curious students may ask a natural question: how was achieved such "purity" of the bifurcation diagrams? They have seen in Fig.2 a)-e) that the fixed and cyclic solutions were not achieved immediately. Initial values of the sequences should splash the diagram. - A rightful doubt, and if students did not ask, the educator has to pose this question by himself.

³Use the following example of syntax: “bifurcation($f_1(x)$, $b+3$,1,4,2)”, where $f_1(x)$ is the recurrence relation ($ax(1 - x)$ in case of logistic map), $b+3$ - the value of bifurcation parameter (a in case of logistic map), 1,4 are the limits of domain: $x \in [1, 4]$, and 2 is the amount of colors.

⁴The line is semitransparent, and these points can be seen through it.

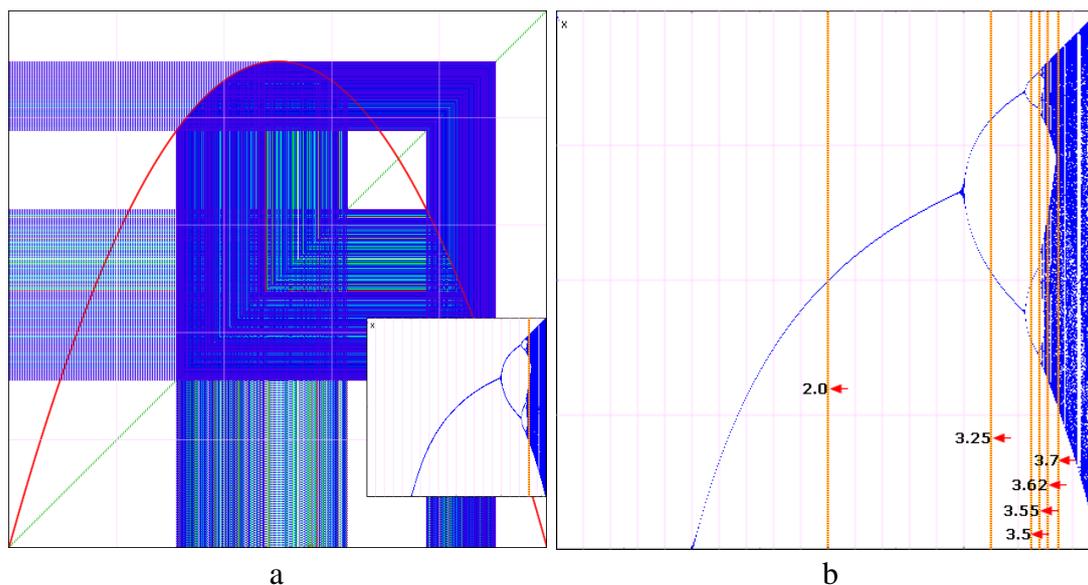


Figure 3

The answer is simple: the diagram shows orbits only after disregarding of the initial, “noisy” elements. Particularly, *VisuMatica* draws x_n only for $n \in [1000, 10000]$. Of course, it never can guarantee a clear image.

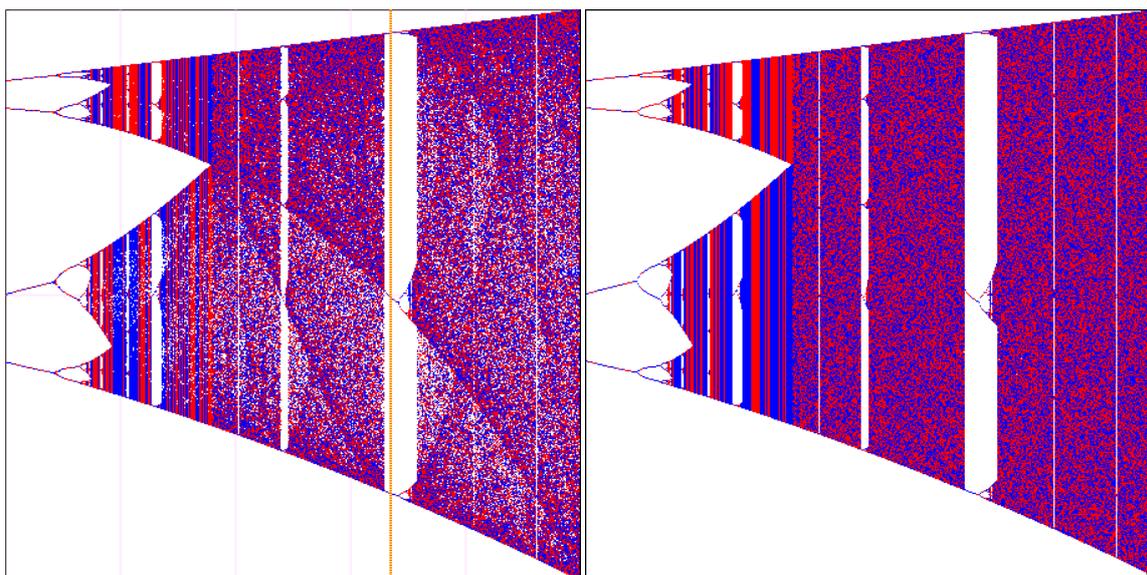


Figure 4 Bifurcation diagram with small (left) and big (right) amounts of iterations

Discussion of correctness of the bifurcation diagram leads us to another subject of examination: there is a suspicious white region approximately around the vertical of 3.9. Of course, it can be a failure of *VisuMatica*’s drawing mechanism, but it worth checking out. After changing the Domain to $[3.5, 4]$ and coloring scheme to two coloring (red, blue)⁵, we receive the bifurcation diagram,

⁵N-coloring pattern in *VisuMatica* paints the sequential elements x_n by the tint with index of remainder of integer division $n \setminus N$.

shown in Fig.4, - an interesting discovery! In this short interval, we have fascinating swapping effects between chaos and order. The white verticals here are holding attractors (3-point attractor in case of $a = 3.835$ is emphasized by yellow-red dotted line), while the painted verticals are presenting chaos. Small change of the bifurcation parameter a dramatically switches the behavior of dynamical system.

It would be useful to discuss with students the reasons of results of coloring with different amount of colors (2, 3, 4), and the visible difference in density distribution in case of chaos and small amount of iterations (left image in Fig.4).

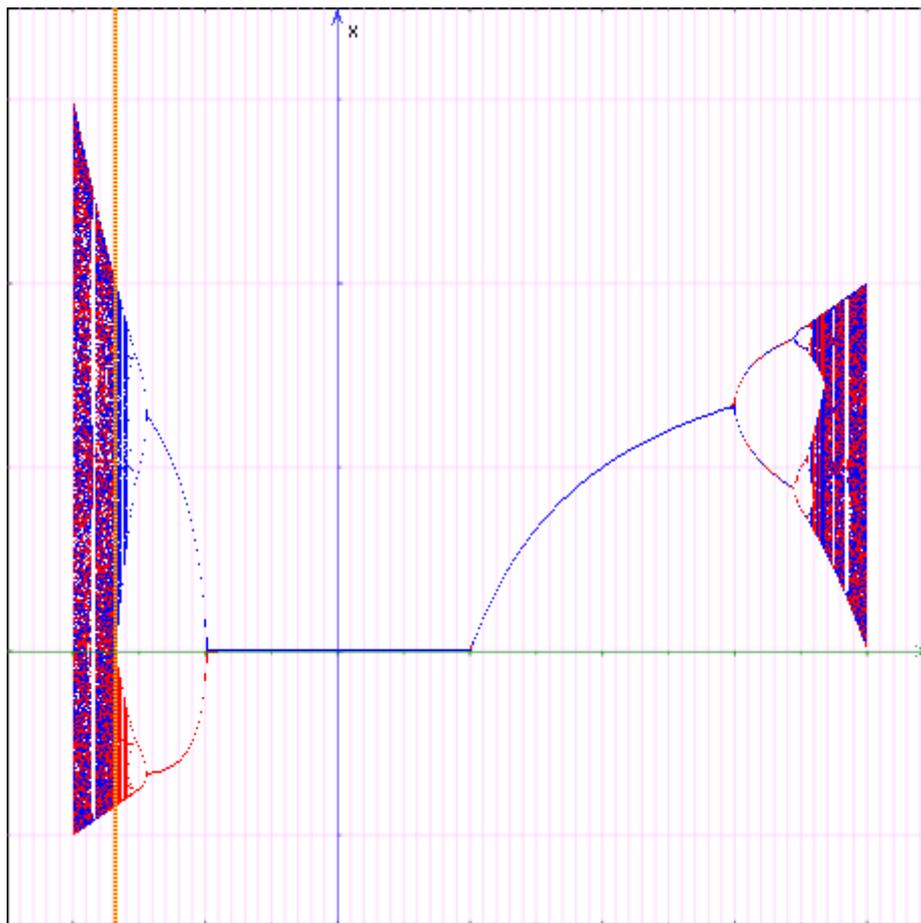


Figure 5

After so fruitful zooming in, on the contrary, let's zoom out from the $[0,4]$ domain, and see what happens in a wider interval of a values. It seems that some extra orbits should be at least in case of $a > 4$. Surprisingly, an observation of Fig.5, where a -domain is $[-2.5, 4.5]$ and x -range is $[-0.75, 1.75]$, leads us to the recognition of the following possibilities in behavior of orbits-solutions of the dynamical system $x_{n+1} = ax_n(1-x_n)$:

- The orbit diverges (for example, in case of $a > 4$).
- The orbit converges to a single value, when $-0.9 < a < 0.9$ and $1 < a < 2.9$.
- The orbit cyclically oscillates between a set of feasible values.
- The orbit behaves chaotically.

The yellow-red line of $a \approx -0.67$ separates an interesting $[-0.67, -1]$ approximate interval of a , where orbits values are sequentially changing from positive to negative vice versa. We see it evidently in the image by the distribution of two-coloring pattern. By the way, similar alternate-hopping behavior can be found in Fig.4 too.

The bifurcation diagram presents a powerful tool in studies of discrete-time dynamical systems. It accumulates the behavioral information overall. A proper selection of number of colors in N-coloring pattern helps to visualize here also the succession of elements in the studied ordered orbits.

It worth to compare with students the pro et contra of presentation of the order in an orbit provided in stair-step diagrams, time series and N-colored bifurcation diagrams.

The outlined progression in introduction and use of these three modeling aids seems didactically preferable. Each one of them has its own unique instructional potential. The teaching logic of educational process causes sequential requests of their application.

2.2. 2D mapping

Another famous example of discrete dynamics delivers the subject of 2D fractals, in particular, the Mandelbrot fractal. We here assume that the reader is familiar with the concept of fractal.⁶

In the situation of no end of software dedicated to fractals, we expect software, which provides the following educational accents:

1. The possibility to visualize the recursive nature of fractal,
2. The possibility to visualize the concrete orbit, which starts at definite location,
3. The possibility to visualize proper objects disclosing the local behavior

VisuMatica supports studies of these features by means of mouse activities and/or dialog based settings and controlling. Thus, the moving manually mouse pointer may be considered as the c parameter of orbit of 0. This orbit, drawn in white (Fig.6) with selectable amount of iterations, helps to explain fractal's coloring. *VisuMatica* shows the orbit as a polyline with segments connecting its elements. Next to the mouse pointer we see the amount of orbit's iterations before it leaves the disk of radius 2,⁷ presented as a surrounding circle. As follows from the definition of \mathbf{M} and drawing

⁶ The *Mandelbrot set* \mathbf{M} is the set of values of c in the complex plane for which the orbit of 0 under iteration of the complex quadratic polynomial $z_{n+1} = z_n^2 + c$ remains inside a closed disk of radius 2 with center at the origin. Thus, \mathbf{M} is a picture in the c -plane, the parameter plane.

The *Julia set* \mathbf{J}_c for the complex quadratic polynomial $f(z) = z^2 + c$ is the set of values of z in the complex plane for which the orbit of z under iteration of the complex quadratic polynomial $z_{n+1} = z_n^2 + c$ remains bounded. Thus, \mathbf{J}_c is a picture in the dynamical plane.

⁷Colors in fractal indicate this amount. Orbits that start at points with the same color have the same amount of iteration before they leave the disk.

algorithm, if point belongs to Mandelbrot set (black area) this number equals to amount of iterations.

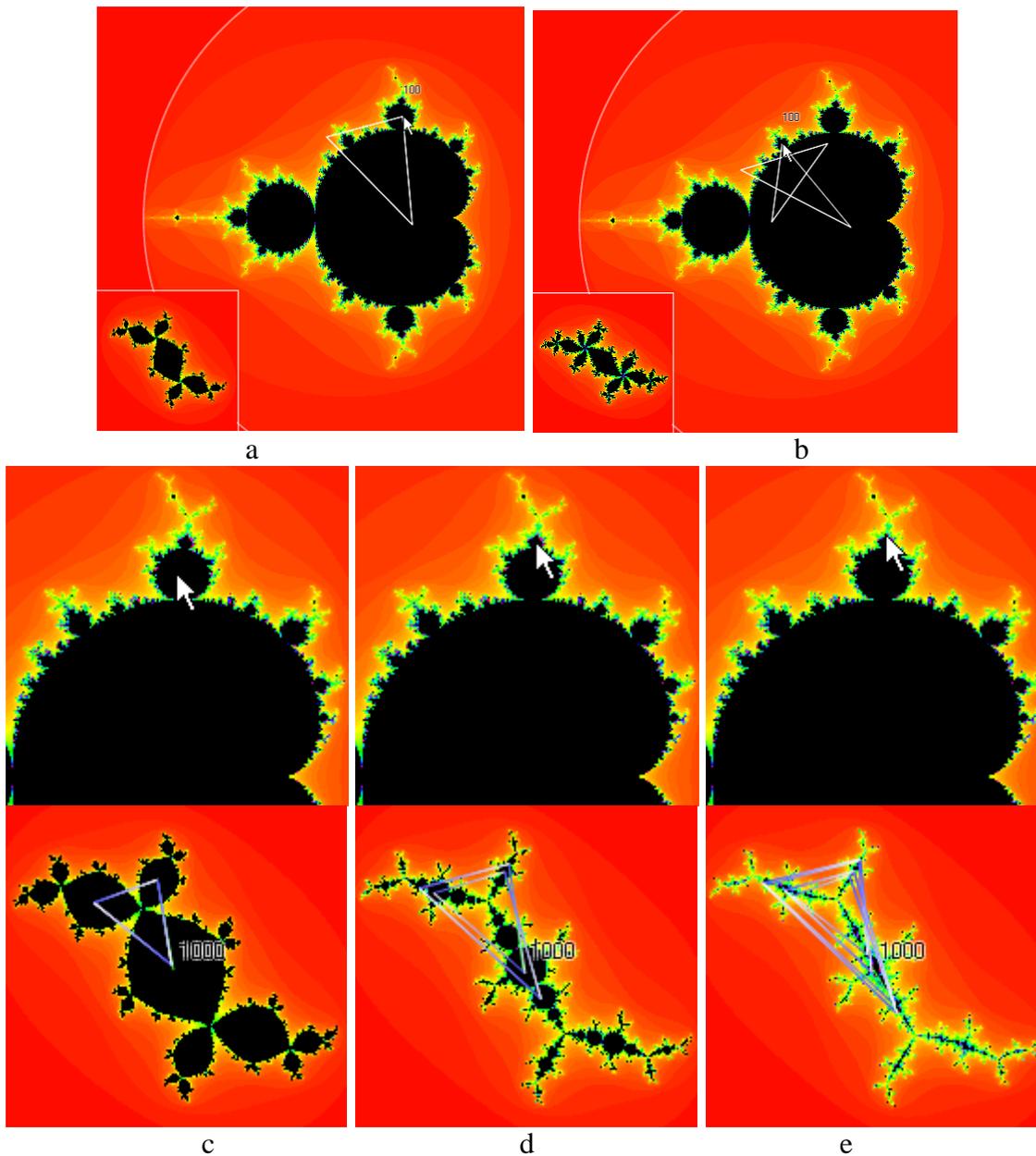


Figure 6

Take into account the multiple bulb-shaped frills directly attached to the main cardioid in M . They called *primary* bulbs or decorations.

In parallel to orbit, VisuMatica allows dynamic view of the Julia set that corresponds to mouse position c in an additional small window. This composite view lets to explore internal features of the Mandelbrot set and its relation to the Julia set (Fig.6 a), b)).

Images in Fig.6 illustrate manual discovery of the characteristic feature of the *bulbs* –their periodicity and their period itself. Namely, if c value (mouse pointed to) is from inside a bulb, then the function

$z^2 + c$ have an *attracting cycle* with some period. Obviously, the *primary* bulb pointed by moving mouse in case a) has period 3, and in case b) – period 5 due to the orbit-polyline. This way the line with 100 segments (default amount of iterations) looks like a closed polygon with 3 and 5 vertices correspondingly.

Appropriate and simultaneous changes of the image of Julia Set in the small left-bottom view validate this conclusion. Really, the amount of black pieces with common joint point in Julia set also equals to the period of the primary bulb. It does not matter what joint point is considered! These pieces appear to be similar and proportional.

Nevertheless, the use of the white orbit, intended to explain fractal's coloring algorithm, for period studies is not surely correct. The accurate and fruitful meaning of the orbit of 0 with mouse pointer, considered as the c parameter, comes when we display the orbit in the Julia subview. The segments now become two-colored. "From white to blue" coloring scheme helps to visualize the order of elements in an orbit.

Such kind of orbit modeling presented by a), b) and c) images in Fig.6. They are aimed to show the fact that entering a secondary bulb in M doubles the period. In general, each advancement to the bulb of next sublevel doubles period of the previous bulb (compare Fig.6 c), d), e)). The Julia set remains connected, but its joints in cases of orbits with initial point in nonprimary bulbs do not express their period.

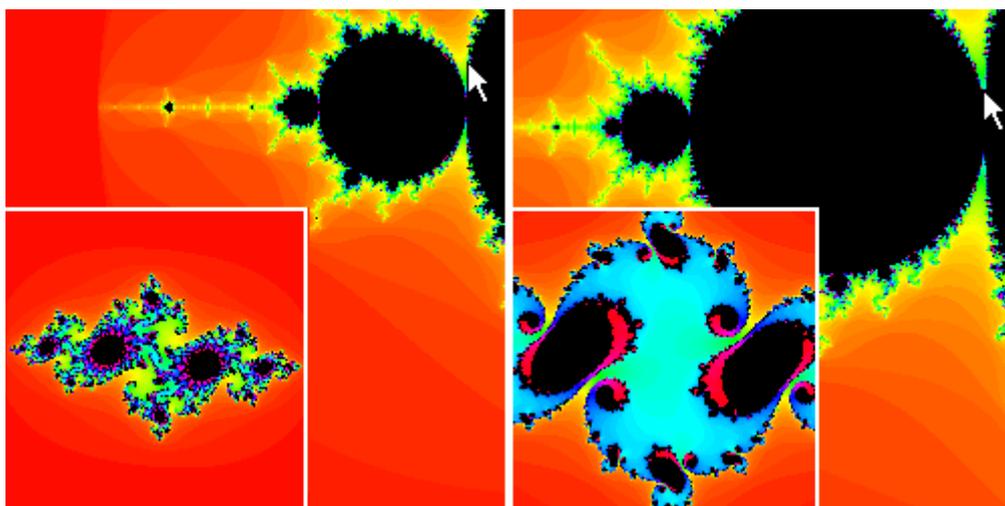


Figure 7

Julia set becomes disconnected when it reflects any point outside of M (Fig.7).

Orbits of internal points of the main cardioid are looking nice, but not expressive (Fig.8 a). The complete trace of all steps of an orbit very often is accompanied with not important information of the starting steps, that disturbs to understand behavior and the tendency of the orbit, especially in case when the orbit attracts to a cycle (Fig.8 b) or, been stable, attracts to some concrete point (Fig.8 a).

VisuMatica allows to omit drawing of the first k steps of the orbit, to connect the initial z_0 with z_k with a green segment and then to connect the rest of elements with directed *from white to blue*

segments $[z_m, z_{m+1}]$, where $k \leq m \leq \text{trace depth}$. This technique cleans the image of the studied orbit and helps to discover its behavior and tendency.

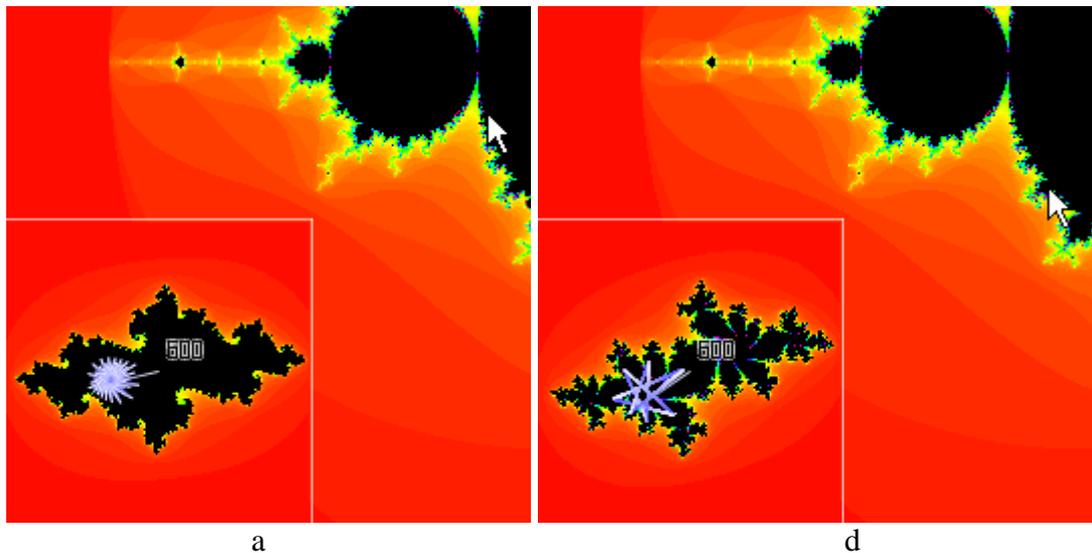


Figure 8

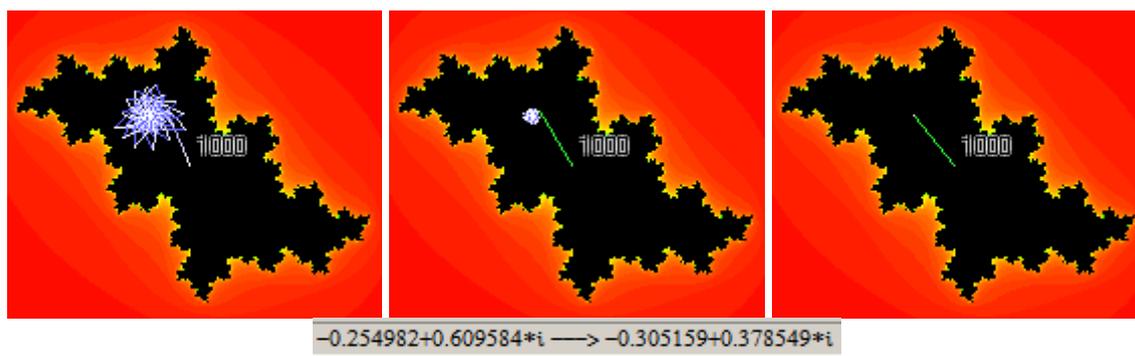


Figure 9

Fig.8 shows three images of an orbit with initial point at $-0.254982+0.609584i$ with the same trace depth of 1000, but with different value of starting index k (0, 50, and 200). The orbit after first 200 steps becomes a point (the attractor). All the rest 9800 segments of the trace are indistinguishable. The status bar at the bottom of *VisuMatica* always displays values of the initial and last iteration points (see it in Fig.9). Both these values remain the same also when we change the depth from 1000 to 500, 10000, 30000, - a very convincing observation.

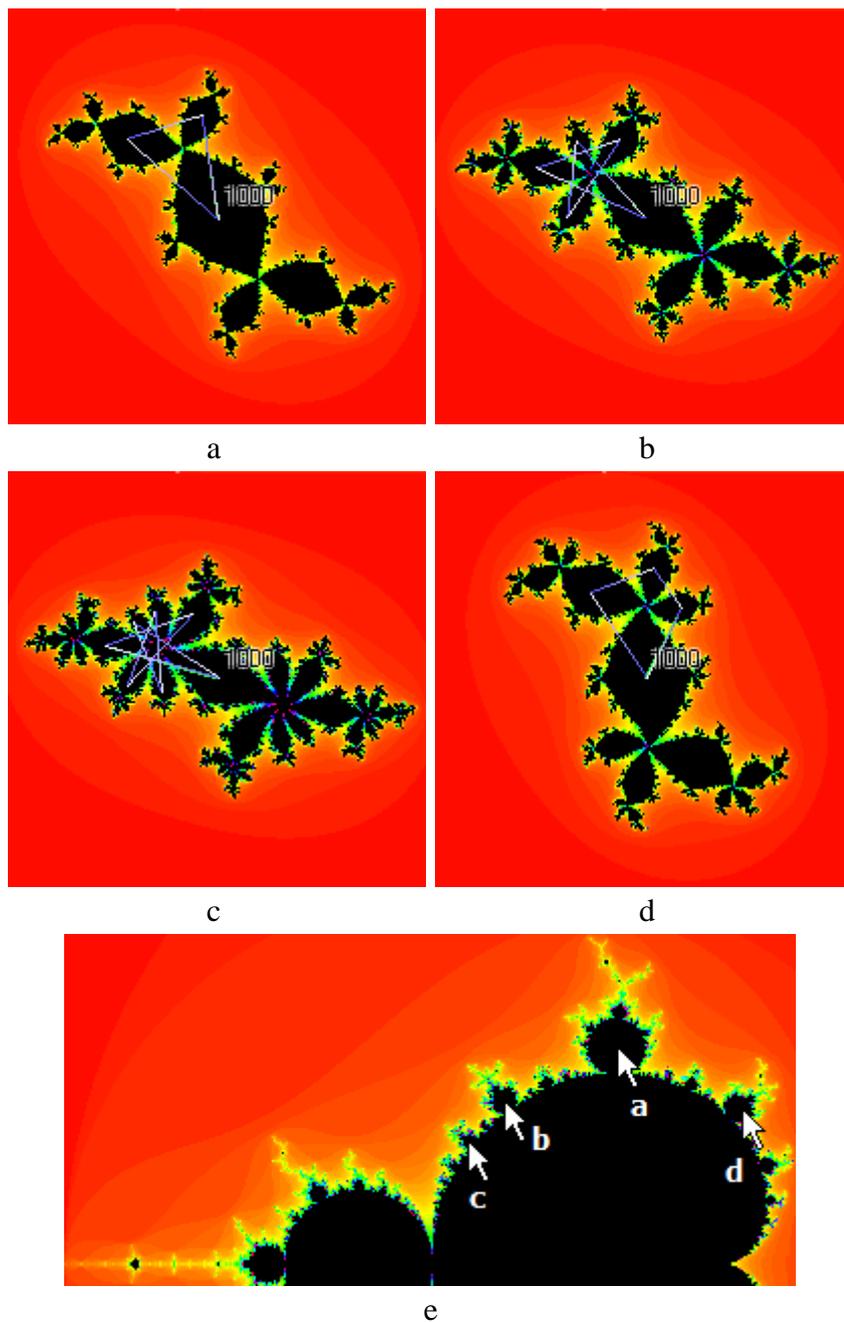


Figure 10

Hereinafter we will keep values of the trace depth and starting index k big enough. The relocation of mouse pointer inside the main cardioid makes clear the fact, that corresponding orbits have period 1, like the point in Fig.9. They get attracted to a fixed complex value and thus are drawn as a single green segment. Inside the bulbs, the orbit becomes a clear p -sided polygon (meaning that the orbit's period equals p). From time to time, this polygon has an accompanied green segment that connects some vertex with the mouse pointer.

Visualizing of precedence by “from white to blue” coloring scheme helps to grasp the value of so called rotation number of a primary bulb, a rational number p/q . We can easily find out numbers p and q by means of orbit-polygon in the small view of Julia set that corresponds to mouse position in

the studied primary bulb. Fig.10 shows images of Julia sets (a) – d)) with orbits that fit mouse locations in the primary bulbs, shown in e). Denominator q is simply the period of the bulb and it equals to amount of vertices in polygon. Thus, q in cases a) – d) equals to 3, 5, 7, and 4 respectively. The attracting cycle hops among q vertices. During this progression, the cycle at each iteration jumps exactly p vertices in the counterclockwise direction from white to blue. Therefore, p in cases a) – d) equals to 1, 2, 3, and 1 respectively, does not matter from which vertex we start counting.

As a challenging task students can discover interesting features of primary bulbs while comparing their periods and rotation numbers.

Consider Julia set itself and its orbits with different initial phase points. Using the technique of omitting drawing of the first orbit's elements, we immediately discover the principal feature of Julia set: the whole set is a basin of attraction; its attractor is a cyclic orbit with elements in the vertices of the unique polygon – the orbit of 0. Fig.11 presents images of J_c , defined with different value of c parameter. We have gathered images of orbits with few initial phase points in the same image to emphasize their common attraction behavior.

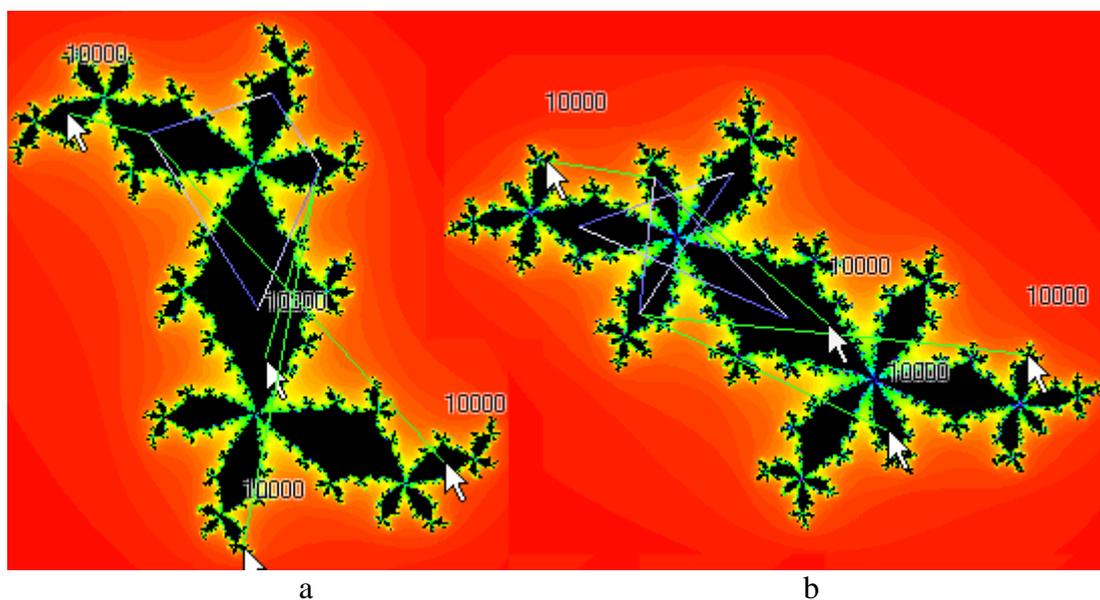


Figure 11

While moving mouse pointer in \mathbf{M} we could detect sudden dramatic changes in the orbit's behavior. We are already familiar with a powerful tool of bifurcation diagram that allows to see the system behavior on the whole. However, in case of complex argument the bifurcation parameter is two-dimensional. For each possible value of parameter the diagram should provide the orbit's sequence of two-dimensional complex numbers as well. Considering its 4D model we obtain some set of points in \mathbf{C}^2 that can not be visualized in a comprehensible manner. So, let us cut it and check the whole picture of what happens *along the real axis* (the “needle”) of the Mandelbrot set with help of 2D bifurcation diagram. In this case the studied function is the real quadratic polynomial $x^2 + c$ for recurrent map $x_{n+1} = x_n^2 + c$ with a real bifurcation parameter c .

Fig.12 presents the Mandelbrot set with a properly located small window of bifurcation diagram. The c -axis defined there in an interval that suits to the Mandelbrot set. The correspondence between

these two images follows from vertical comparison. Thus, the yellow-red dotted vertical line in the diagram fits $c = -1.25$. Here system traverses from the primary bulb with period 2 to to the “derived” secondary bulb with period 4.

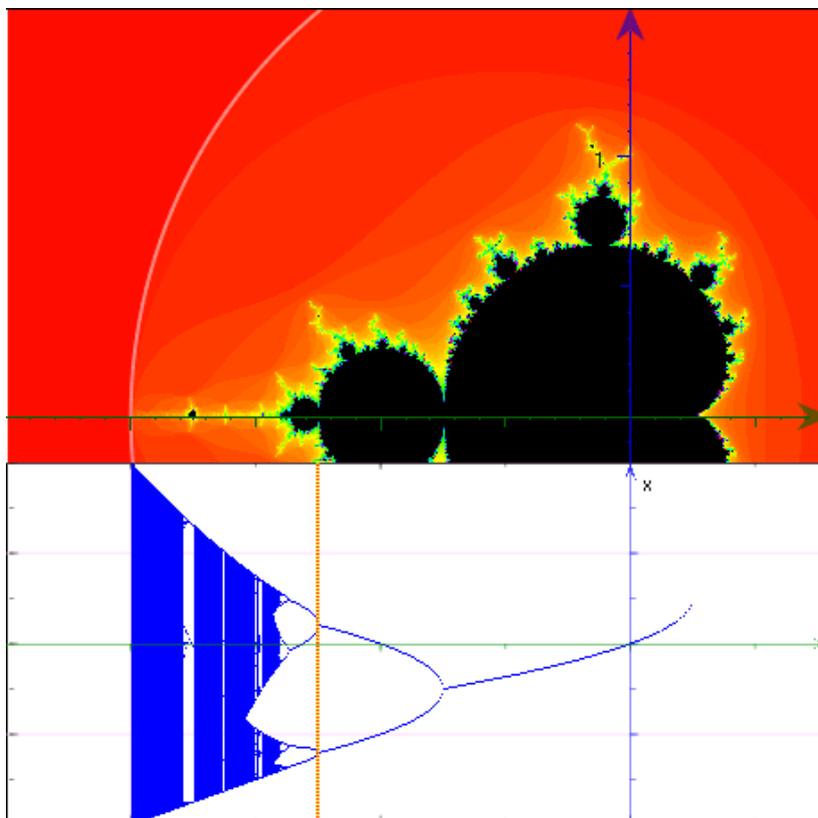


Figure 12

The map looks similar to the bifurcation map of logistic system. So, students can explore it in the same way as previously. In particular, they can change value of c and thus find out the period changes and effects of swapping between chaos and order. They are also familiar with the technique of zooming into a suspicious white area. Now students can take advantage of the fractal’s point-and-click zooming mechanism: it zooms both models equally (Fig.13). One more helpful feature: *VisuMatica* constantly presents current coordinates of the moving mouse pointer.⁸ So, one can consider the abscissa in this pair as the value of bifurcation parameter c and click the mouse button in a proper point of the image. This way we have discovered the proper c -parameter value - 1.777228 of the yellow-red vertical line in Fig.13 (pay attention to the image of cursor and the fragment of status bar at the bottom). In this regard is very appropriate a question on the reasons and location of the second complex number in the status bar.

⁸In case of fractals, it can do more: it presents the mouse location in notation of complex numbers. Additionally, *VisuMatica* shows the value of correspondent orbit’s last iteration element.

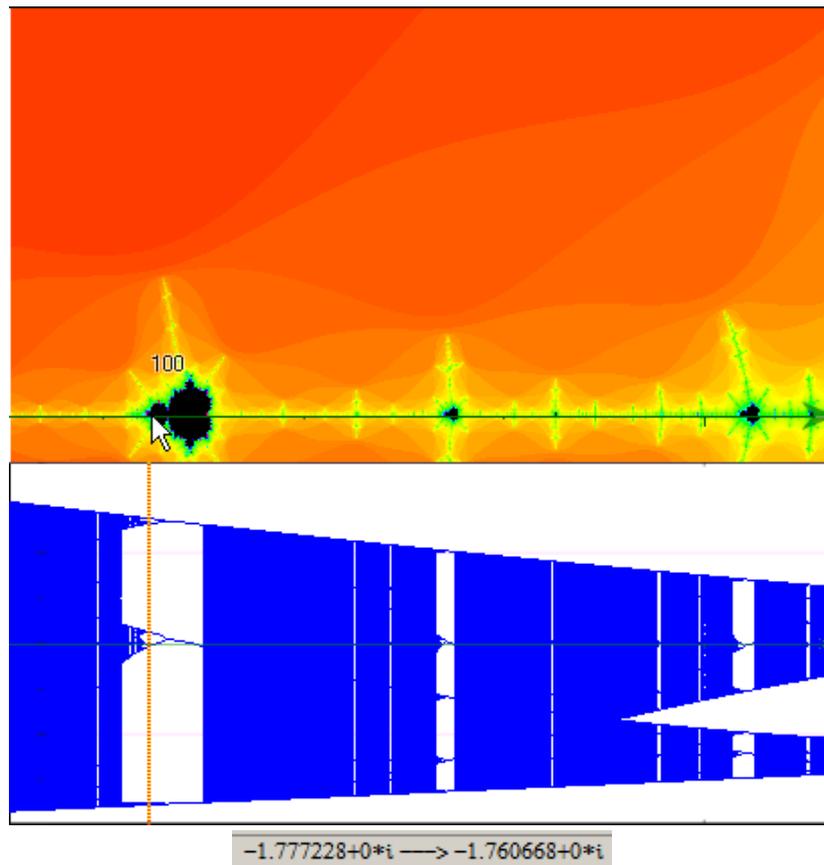


Figure 13

The zooming mechanism has few options:

- The caller - clicked point - remains in the same location on the screen (*natural zoom*).
- The caller - clicked point - becomes the center of the zoomed image (*central zoom*).

Fig.13 was produced by the technique of *natural zooming* (click some point on the real axis while holding the “*Shift*” button on the keyboard).

Let observe Fig.12 once more. On the left of main cardioid sits the primary bulb with period 1. The period of each next derived bulb on the left is twice bigger. All the bulbs are looking as circles⁹. Points of their tangency are points of period doubling bifurcation. All these points lie on the real axis (one of them emphasized in Fig.12). Their abscissas form some sequence. It is not difficult to discover approximate values of its elements using central zoom and proper mouse location. Being decreasing and bounded from below, the sequence converges.

Studies of this sequence and derived from it sequences of bulbs’ diameters (as differences of successive values) and their ratios leads students to discovery of generality between Mandelbrot and logistic non-linear discrete mapping and its expression by common *Feigenbaum* constant.

It should be noted that the location of mouse pointer can be adjusted exactly or controlled dynamically by means of the simulation mechanism, which was used in studies of Zhukovsky transformation.

⁹ They are circles.

2.3. 3D mapping

In studies of 3D nonlinear continuous dynamical systems, the orbit view as a spatial curve not always informative enough. *Like a sculpture, we observe the orbit from different points of view (instead of moving around the sculpture, we rotate the 3D volume with help of the software interface). Unlike a sculpture; such observation may be insufficient because of complexity of internal structure of the 3D curve. The regular way to study internal structure of some object consists in an analysis of its cuts.*

Poincaré section considers points of intersection of the orbit with a transversal plane. Thus, the Poincaré section reduces a continuous 3D flow to a discrete-time¹⁰ 2D mapping.

VisuMatica presents these points as colored small balls and outlines the section plane in grey (Fig.14). Fig.14 e) shows interface of the featuring dialog. One can select:

- The maximum allowed amount of intersection points (depth).
- Whether to take into account points of intersection only in case of orbits' hits from "positive" side of the section plane (used in Fig.14), from the "negative" one, or from both sides.
- Show arcs of orbit in the suitable half-space with different colors or draw orbit in a single color.

Colors of balls and arcs encode the sequence of intersections in rainbow order: from red to violet. To view the ordinal number of a point in Poincaré section, user indicates the point by mouse (see Fig.14 a), b)).

The speed of drawing in VisuMatica, like in any other software, strictly depends on the precision of calculations. In order to achieve quick "real-time" performance, developers reduce the accuracy of the calculations. In case of dynamical systems it may lead to serious mistakes in the resulting image and in conclusions, which are based on it.

Fig.14 presents four images of Poincaré maps obtained with different values of depth (10 in case a) and 100 in cases b) – d)), and *controllable precision* (low - a), b), medium c), high d)).

So, more or less correct image of orbit suspicious to be a periodic cycle is shown in Fig.14 d). The intersection points (all the 100 colored balls) coincide. It worth to *pay students' attention to the rainbow-like color distribution in cases a) – c) and to discuss the reasons of such coloring and the intersections' shrinkage effect.*

¹⁰The time interval from point to point is not surely constant here.

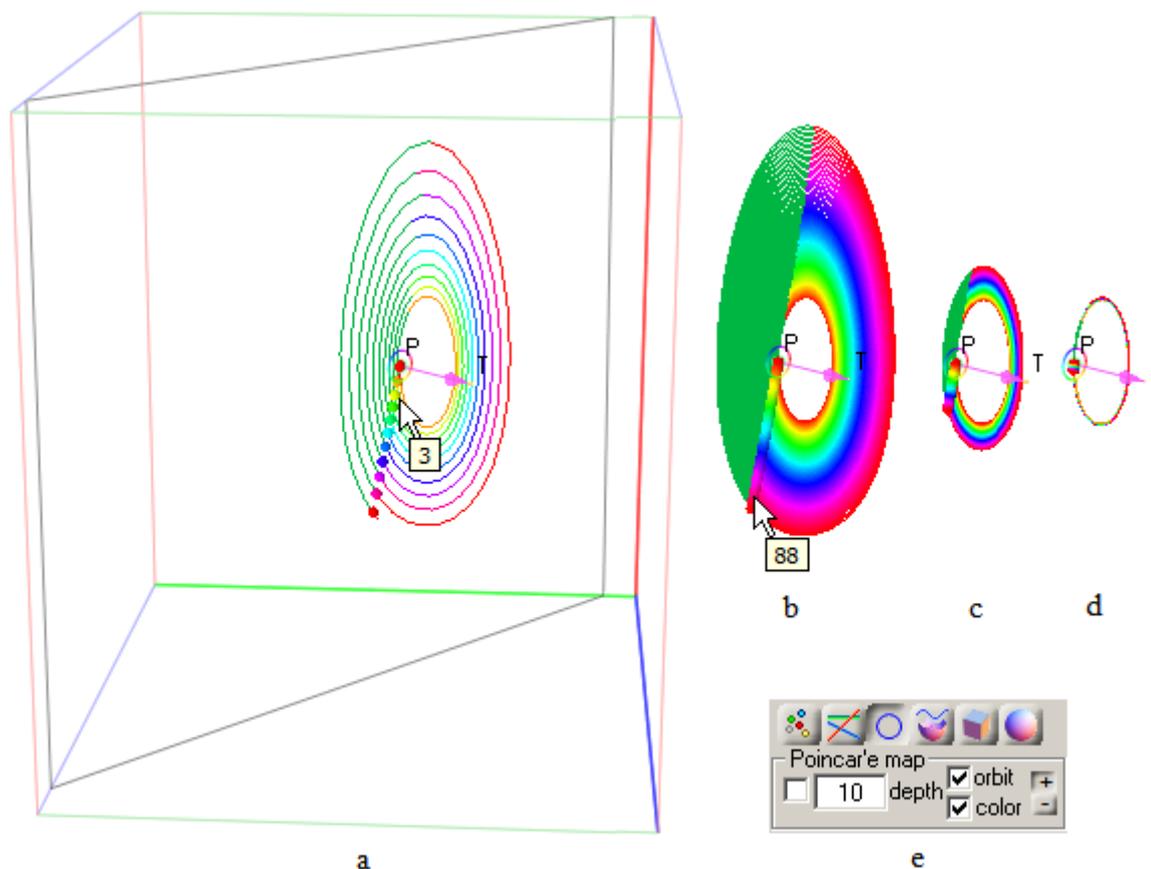


Figure 14

User defines the plane of Poincaré section as a plane of some circle. It eases manual activities with the plane: we can exploit the existing interface of manipulations with circles. There are two ways of circle definition that are suitable for this goal: a circle, whose plane is normal to the orbit that passes through its center, and a circle, which is defined by center and a normal vector to its plane (Fig.14). The problem of precision arises when user tries to redefine the plane manually. To solve it *VisuMatica* automatically reduces precision of calculations to prevent screen's "freezing" and restores setting, when user releases the mouse button.

At first sight, something similar to the first image in Fig.14 is displayed in Fig.15, which depicts the orbit of Lorenz attractor: the single spiral just becomes a dual one. The balls–points of intersection are also similarly located close one to another. But the difference is much more serious, and it is easily recognizable in the not regular distribution of colors in Lorenz attractor. Increasing of the calculations precision does not lead to effect of orbit shrinkage we have seen in case of periodic cycle. The strange behavior of Lorenz attractor manifested in the fact that it never repeats and chaotically proceeds: two close points on the attractor become randomly far away from each other later.

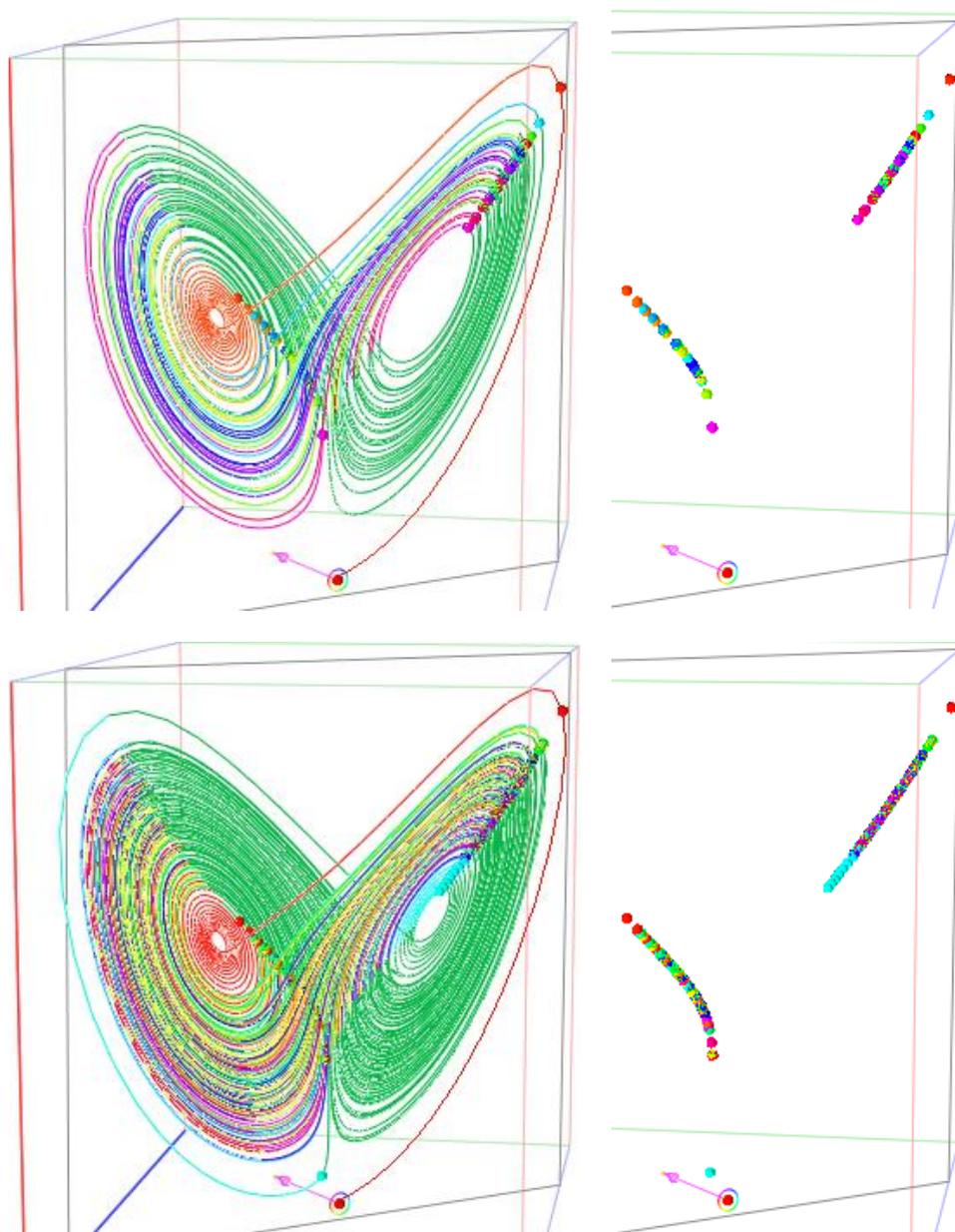


Figure 15 Amount of intersections: upper row – 50, lower row – 200.

Let us slightly perturb initial point of the orbit and instead of coordinates $(1.0, 0.01, 0.48)$ - left column in Fig.16 - assign a new triplet: $(1.00000001, 0.01, 0.48)$ – right column in Fig.16. The perturbation is really light: we've added 0.00000001 to x -coordinate.¹¹

We receive surprisingly different result. Everything have changed: orbit, intersection points, and distribution of colors. The lower row in Fig.16 shows exclusively the intersection points – the Poincaré mapping, which is coded geometrically as location of points supplemented with visual colored coding of their sequence. The geometry here is rather similar, but the color difference tells

¹¹ All the images in Fig.15 have 100 intersection points and produced with 50000 iterations.

the true story about chaotic behavior: these two orbits at the same time t are very far from each other.

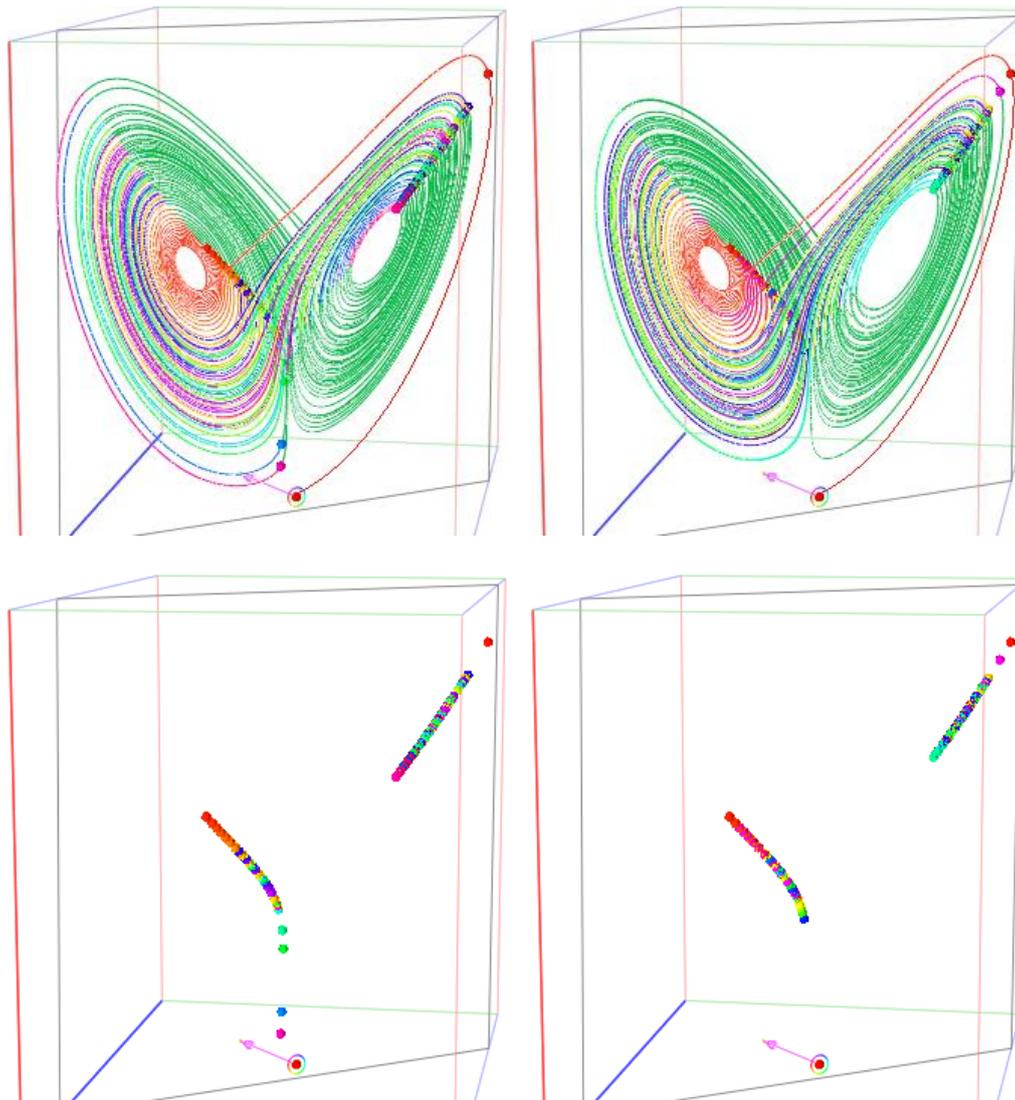


Figure 16

Fig.17 shows two ways of confirmation of the noticed difference:

- Case e) presents evolution of a segment that connects these two points.
- Images a) – d) are displaying few stages of evolution of a circle of radius 0.000001 with center in the initial point of attractor and located in its normal plane.

Each point on the circle, being initial phase point of some orbit, lie so close to the initial point of the attractor that we can not distinguish them (0). After definite time interval the image of the circle becomes visible (1). Sometimes it grows (2), (6), (7), sometime it shrinks (3), (4), (5) remaining smooth. But some moment, the previously very close orbits start move away (pink segments in 12) and the image of the circle splits into smooth curves connected with unicolorous segments (12), (13). Later, these smooth images of arcs become indistinguishable and the circle looks as a

chaotically deforming closed polyline with randomly distributed vertices, which are moving on their orbits independently and smoothly (14).

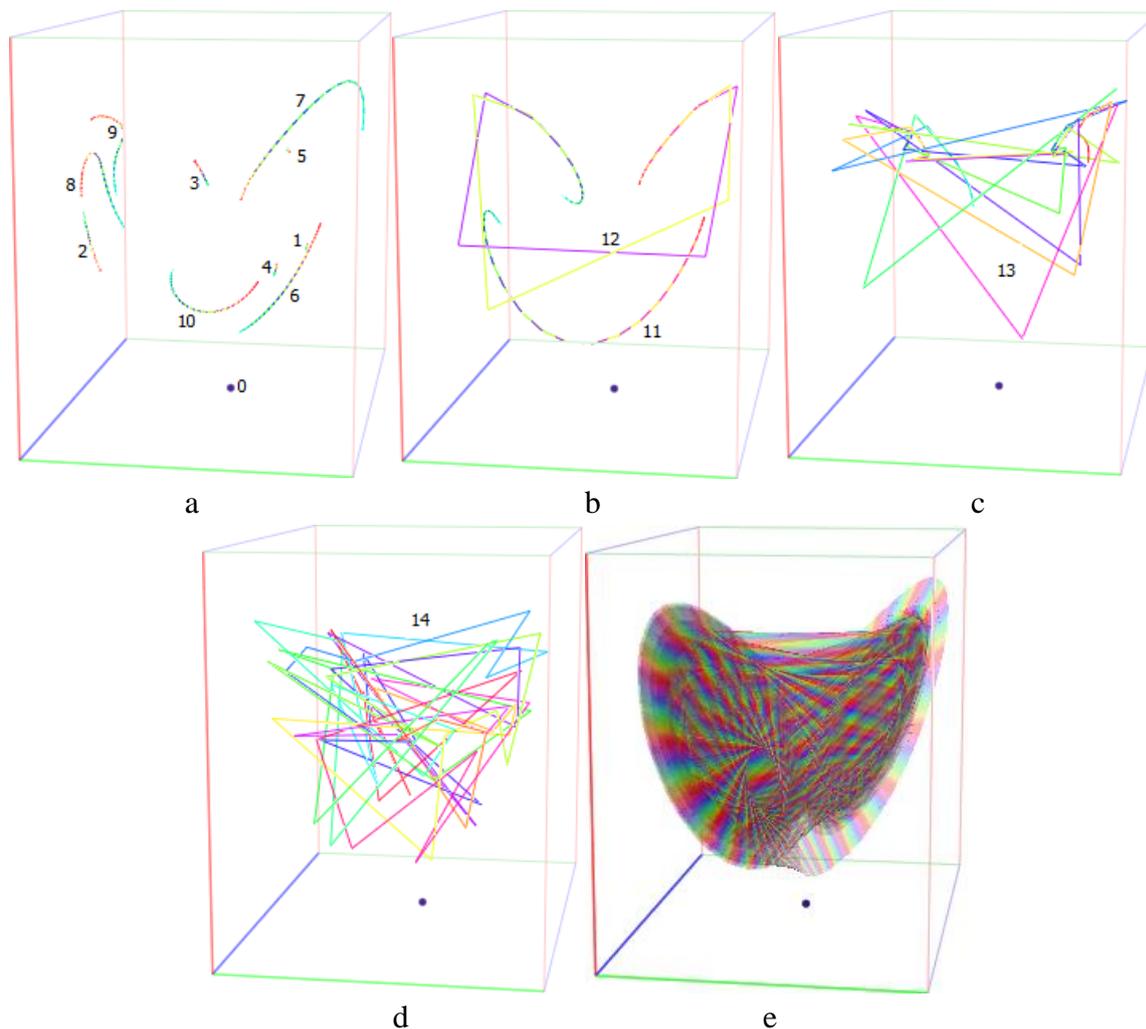


Figure 17

Suggestive questions, like:

- *Is (14) really a polyline?*
- *Should we expect changes of the amount of vertices?*

and other comparison activities help students to grasp the idea of **butterfly effect**: a sensitive dependency on initial conditions in which a small change at one place in a deterministic nonlinear system can result in large differences in a later state.

3. Conclusions

This paper demonstrates a new way in software support of teaching and learning mathematics, in particular, nonlinear space transformations.

Unlike applets and specific models in CAS and DG systems, here creation and setting mechanism have a generic nature. This approach allows emphasize and exploit the multidisciplinary and

interdisciplinary links, the unity of mathematics, and thus to ease study of different mathematical subjects.

Distinguishing features of VisuMatica:

- Absence of special syntax for definition and configuration of mathematical objects,
- Intensive use of the powerful potential of user interface,
- Attainment of dynamicity by manual manipulations with mouse and sliders,
- Information visualization by colored coding

help students to explore complicated mathematical notions and get insight about their properties and relationships.

Analysis of the visual artefacts pays students attention to the limitations of the software and hardware processing, fosters criticality in evaluating the computing results.

References

- [1] Arnold V.I. Ordinary differential equations. The MIT Press, 1978
- [2] Bertuglia C. S., Vaio F. Nonlinearity, chaos & complexity. The dynamics of natural and social systems. Oxford University Press, 2005
- [3] Devaney R. L. The Fractal Geometry of the Mandelbrot Set. http://is.muni.cz/el/1456/jaro2009/PMAPEM/The_Fractal_Geom/The_Fractal_Geometry_Of_The_Mandelbrot_Set.html
- [4] Enns R. H. It's a Nonlinear World, Springer, 2010
- [5] Glendinning P. Stability, Instability and Chaos. An Introduction to the Theory of Nonlinear Differential Equations. Cambridge University Press, 1994
- [6] Katz G., Nodelman V. The Shape of Algebra in the Mirrors of Mathematics, World Scientific, 2012
- [7] Katz G., Nodelman V. Software Tools for Visualizing Multivalued Function. Research Journal of Mathematics & Technology, Vol. 2, Num. 1, 2013
- [8] Marotto F. R. Introduction to Mathematical Modeling Using Discrete Dynamical Systems. Thomson Brooks/Cole, 2006
- [9] Wagon S. New Visualization Ideas for Differential Equations. <http://www.cecm.sfu.ca/organics/papers/wagon/paper/html/wagon.html#wagon15>
- [10] Weisstein, Eric W. Conformal Mapping. <http://mathworld.wolfram.com/ConformalMapping.html>