

УДК: 511.11, 519.677, 517.542

Экспериментальное исследование динамики одинокных и связанных в решетке комплекснозначных отображений: архитектура и интерфейс авторской программы для моделирования

И. В. Матюшкин^{1,a}, П. Д. Рубис¹, М. А. Заплетина^{2,b}

¹Национальный исследовательский университет «Московский институт электронной техники»,
Россия, 124498, г. Москва, г. Зеленоград, Площадь Шокина, д. 1

²Институт проблем проектирования в микроэлектронике РАН,
Россия, 124365, г. Москва, г. Зеленоград, ул. Советская, д. 3

E-mail: ^a imatyushkin@niime.ru, ^b zapletina_m@ippm.ru

Получено 11.10.2021.

Принято к публикации 02.11.2021.

В работе описывается свободно распространяемая прикладная программа для исследований в области голоморфной динамики на основе вычислительных возможностей среды MATLAB. Программа позволяет строить не только одиночные комплекснозначные отображения, но и их коллективы как линейно связанные, на квадратной или гексагональной решетке. В первом случае строятся аналоги множества Жюлиа (в виде точек убегания с цветовой индикацией скорости убегания), Фату (с выделением хаотической динамики) и множества Мандельброта, порожденного одним из двух свободных параметров. Во втором случае рассматривается только динамика клеточного автомата с комплекснозначным состоянием ячеек и всеми коэффициентами в локальной функции перехода. Абстрактность объектно-ориентированного программирования позволяет объединить оба типа расчета в рамках одной программы, описывающей итеративную динамику одного объекта.

Для формы поля, начальных условий, шаблона окрестности и особенностей окрестности у граничных ячеек предусмотрены опции выбора. Вид отображения может быть задан регулярным для интерпретатора MATLAB выражением. В статье приводятся некоторые UML-диаграммы, краткое введение в пользовательский интерфейс и ряд примеров.

В качестве рабочих иллюстраций, содержащих новое научное знание, были рассмотрены следующие случаи:

1) дробно-линейное отображение вида $Az^n + B/z^n$, для которого случаи $n = 2, 4, n > 1$, известны. На портрете множества Фату привлекают внимание характерные (для классического квадратичного отображения) фигурки «пряничных человечков», показывающие короткопериодические режимы, находящиеся в море компоненты условно хаотической динамики;

2) у множества Мандельброта при нестандартном положении параметра в показателе степени $z(t+1) \leftarrow z(t)^n$ на эскизных расчетах обнаруживаются некие зубчатые структуры и облака точек, напоминающие пыль Кантора, не являющиеся букетами Кантора, характерными для экспоненциального отображения. В дальнейшем требуется детализация этих объектов со сложной топологией.

Ключевые слова: голоморфная динамика, нелинейная динамика, клеточные автоматы, экспериментальная математика, фракталы, MATLAB, итерированные отображения

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований: проект РФФИ № 20-07-00409.

UDC: 511.11, 519.677, 517.542

Experimental study of the dynamics of single and connected in a lattice complex-valued mappings: the architecture and interface of author's software for modeling

I. V. Matyushkin^{1,a}, P. D. Rubis¹, M. A. Zapletina^{2,b}

¹National Research University «Moscow Institute of Electronic Technology»,
1, Skokin Square, Zelenograd, Moscow, 124498, Russia

²Institute for Design Problems in Microelectronics of Russian Academy of Sciences,
3 Sovetskaya st., Zelenograd, Moscow, 124365, Russia

E-mail: ^a imatyushkin@niime.ru, ^b zapletina_m@ippm.ru

Received 11.10.2021.

Accepted for publication 02.11.2021.

The paper describes a free software for research in the field of holomorphic dynamics based on the computational capabilities of the MATLAB environment. The software allows constructing not only single complex-valued mappings, but also their collectives as linearly connected, on a square or hexagonal lattice. In the first case, analogs of the Julia set (in the form of escaping points with color indication of the escape velocity), Fatou (with chaotic dynamics highlighting), and the Mandelbrot set generated by one of two free parameters are constructed. In the second case, only the dynamics of a cellular automaton with a complex-valued state of the cells and of all the coefficients in the local transition function is considered. The abstract nature of object-oriented programming makes it possible to combine both types of calculations within a single program that describes the iterated dynamics of one object.

The presented software provides a set of options for the field shape, initial conditions, neighborhood template, and boundary cells neighborhood features. The mapping display type can be specified by a regular expression for the MATLAB interpreter. This paper provides some UML diagrams, a short introduction to the user interface, and some examples.

The following cases are considered as example illustrations containing new scientific knowledge:

1) a linear fractional mapping in the form $Az^n + B/z^n$, for which the cases $n = 2, 4, n > 1$, are known. In the portrait of the Fatou set, attention is drawn to the characteristic (for the classical quadratic mapping) figures of «gingerbread men», showing short-period regimes, components of conventionally chaotic dynamics in the sea;

2) for the Mandelbrot set with a non-standard position of the parameter in the exponent $z(t+1) \leftarrow z(t)^\mu$ sketch calculations reveal some jagged structures and point clouds resembling Cantor's dust, which are not Cantor's bouquets that are characteristic for exponential mapping. Further detailing of these objects with complex topology is required.

Keywords: holomorphic dynamics, nonlinear dynamics, cellular automata, experimental mathematics, fractal, MATLAB, iterated mappings

Citation: *Computer Research and Modeling*, 2021, vol. 13, no. 6, pp. 1101–1124 (Russian).

The work was supported by RFBR, project number 20-07-00409.

1. Введение

Цель статьи состоит в презентации авторской программы Complex Mappings Researcher (CMR), ориентированной на профессиональных математиков, работающих в области голоморфной динамики и комплекснозначных клеточных автоматов. Программа CMR, созданная на платформе MATLAB, является свободно распространяемой по лицензии MIT и доступна для скачивания [Github-репозиторий программы. . . , 2021]. Итерированная нелинейная динамика и статичные фрактальные образы, потенциально генерируемые CMR, служат прекрасными примерами приложения идей экспериментальной математики [Арнольд, 2005; Experimental mathematics, 2021]. В данном случае компьютер выступает инструментом эвристики и верификации каких-либо математических гипотез относительно отображений вида $z \rightarrow \varphi(z)$, $z \in \mathbb{C}$ или $Z \rightarrow \varphi(Z)$, $Z \in \mathbb{C}^n$; последний объект трактуется нами как клеточный автомат (КА), значение ячейки которого есть комплексное число, или как решетка связанных отображений (coupled map lattice, [Kaneko, 1992; García-Morales, 2016]). Абстрактность объектно-ориентированного программирования, для которого сложность изменяющегося по неким правилам объекта не принципиальна, позволяет реализовать в одной программе исследование обоих объектов. В текущей версии программы КА предполагается двумерным и заданным на квадратной или гексагональной решетке. Исследование связанных отображений [Кулаков, Фрисман, 2015; Поздняков, 2011; Feudel, 2008] соответствует подходу коннекционизма, актуальности исследования феноменов нелинейной динамики в мультиагентных системах, примитивным примером которых является система осцилляторов Курамото [Miritello et al., 2009].

Предыстория CMR основана на визуализации топологии букетов Кантора у множеств Жюлиа и Мандельброта для экспоненциального [Матюшкин, 2016] и линейно-экспоненциального отображений [Матюшкин, 2016; Matyushkin, Zapletina, 2018]. Ряд методических особенностей эксперимента указан нами в [Матюшкин, Заплетина, 2018]; например, множество Жюлиа отождествляется с множеством точек убегания.

Структура данной статьи следующая. В п. 1 мы рассмотрим список и математические формулировки задач, а также возможности их спецификации; затем в п. 2 кратко опишем технические параметры программы и особенности программирования, а также приведем основные UML-диаграммы. В п. 3 приведем ряд примеров-демонстраций работы программы. В п. 4 отразим некоторые интересные результаты для двухпараметрических семейств отображений вида $\mu z + \mu_0 z^{-1}$, $\mu_0 + z^\mu$.

2. Математический базис программы CMR

Программа Complex Mappings Researcher помогает исследовать следующие объекты.

1. Произвольные, но задаваемые в виде интерпретируемых формул, итерированные комплекснозначные функции вида (1), где $z(t+1)$ — значение итерированной функции $F(\mu, \mu_0, z(t))$ на итерации $t+1$:

$$\begin{aligned} z(t+1) &= F(\mu, \mu_0, z(t)), \\ z \in \mathbb{C}, \quad t &= 0, 1, 2, \dots, \quad \mu, \mu_0 \in \mathbb{C} \text{ — параметры.} \end{aligned} \quad (1)$$

2. Библиотечные функции мультипликативного вида «модификатор – база» (2) с предопределенными сценариями (3):

$$z(t+1) = \lambda(\mu, \mu_0, z(t)) \cdot base(z(t)), \quad (2)$$

$$\begin{aligned} base: \quad & 1, \quad z^2 + \mu, \quad \exp(iz), \\ \lambda: \quad & \mu + \mu_0, \quad \mu + z, \quad \mu + \mu_0|z|, \quad \mu + \mu_0(z - z^*), \quad \mu + \mu_0|z - z^*|. \end{aligned} \quad (3)$$

Константа z^* порождена базой $base = \exp(iz)$, являясь ближайшим к нулю корнем уравнения $z = \mu \cdot \exp(iz)$ и задавая неподвижную точку; в остальных случаях $z^* = 1 + 0i$.

3. Произвольные КА с локальной функцией перехода (ЛФП) вида (4), зависящей от двух внешних параметров (μ, μ_0) , состояния центральной ячейки z и ее веса μ_c , состояний k соседей $\langle z_1, \dots, z_K \rangle$, $K \leq 8 \in \mathbb{N}$ и K параметров связи $\langle \mu_1, \mu_2, \dots, \mu_K \rangle$:

$$z_\alpha(t+1) = F(\mu, z_\alpha(t), \{z_{\alpha+\beta}(t)\}), \quad \alpha - \text{индекс}, \quad 1 \leq \beta \leq K - \text{сдвиг}, \quad (4)$$

$$Z = \{z_\alpha\}, \quad \mu - \text{параметры}, \quad \dim \mu = K + 3.$$

Здесь $z_\alpha(t)$ — состояние центральной ячейки окрестности на итерации t . Весовые коэффициенты определяют степень связанности и, соответственно, влияние соседей-осцилляторов на центральную клетку.

4. Библиотечные функции ЛФП вида (5) с предопределенными сценариями (6):

$$z_\alpha(t+1) = \lambda(\mu, z_\alpha(t), \{z_{\alpha+\beta}(t)\}) \cdot base(z_\alpha(t)), \quad (5)$$

$$base: \quad 1, \quad z_\alpha^2 + \mu, \quad \exp(iz_\alpha),$$

$$\lambda: \quad \mu_0 + \sum_{k=1}^K \mu_k z_k, \quad \mu + \mu_0 \left| \sum_{k=1}^K (-1)^j z_k \right|, \quad \mu + \mu_0 \left| \frac{1}{K} \sum_{k=1}^K z_k - z^*(\mu) \right|, \quad \mu + \mu_0 \left(\frac{1}{K} \sum_{k=1}^K z_k - z^*(\mu) \right). \quad (6)$$

Константа z^* определяется аналогично из равенства $z^* = \mu \cdot \exp(iz^*)$ и соответствует глобальному равновесию КА в однородной конфигурации ($\forall \alpha$) ($z_\alpha = z^*$).

5. Клеточные автоматы с двумя вариантами геометрии поля: квадрат и гексагон (как вертикальный, так и горизонтальный). При квадратной форме поля форма ячейки представляет собой квадрат либо гексагон (по умолчанию вертикальный). Если форма поля — горизонтально расположенный гексагон, то форма ячейки — вертикальный гексагон, и наоборот. Для обеих геометрий поля вводятся шаблоны окрестности Мура и Неймана, причем каждый сосед получает геометрически детерминированный номер от 1 до K , используемый в аргументах ЛФП.

Для граничных ячеек КА предусмотрен особый вид ЛФП по одному из трех сценариев: 1) «линия смерти», когда состояние ячейки не изменяется; 2) с замыканием, когда окрестность дополняется соседями с противоположного конца поля; 3) изолированный, когда исчисление ЛФП подстраивается под фактическое число соседей, что легко реализовать для библиотечных ЛФП.

Выбор начальных условий $Z(0)$ зависит от индексов (x, y) ячейки КА внутри поля и предполагает один из четырех сценариев распределений, параметризуемых тремя величинами $\langle a, b, c \rangle$ и реперным значением z_0 :

- 1) линейное распределение: $z(x, y) = z_0 + a + bx + cy$, $0 \leq x, y < M$ (или $3N$);
- 2) центрально-симметричное линейное: для **гексагонального** поля $z(x, y) = z_0 + a + Rn(b + c)$, где Rn — номер концентрического гексагона внутри КА-поля, которому принадлежит ячейка с координатами x, y ; для **квадратного** поля $z(x, y) = z_0 + a + b \left| x - \frac{M-1}{2} \right| + c \left| y - \frac{M-1}{2} \right|$;
- 3) случайное равномерное: $z(x, y) = z_0 + a + (b-a)p_1 + ic[a + (b-a)p_2]$, $b > a$, $\text{Im } a = \text{Im } b = \text{Im } c = 0$, $p_{1,2} \in [0; 1]$, где $p_{1,2}$ — результаты двух испытаний для равномерного распределения;
- 4) случайное нормальное: $z(x, y) = z_0 + c(p_1(a, b) + ip_2(a, b))$, $\text{Re}(b) > 0$, $\text{Im } a = \text{Im } b = \text{Im } c = 0$, $p_{1,2}: f = \frac{1}{b\sqrt{2\pi}} \exp\left(-\frac{(x-a)^2}{2b^2}\right)$, где $p_{1,2}$ — результаты двух испытаний для гауссова распределения формы $f(a, b)$.

Частные замечания о геометрии КА

Для полей квадратной формы $M \times M$ используется две системы координат: глобальная, где левой нижней ячейке соответствуют координаты $(x, y) = (0, 0)$, $x, y < M$ (правая система, соответствующая зубчатому массиву, где строки располагаются одна над другой, а нулевая абсцисса соответствует левому концу строки) и локальная, где соседи обходятся по часовой стрелке (рис. 1, а, б). Для поля гексагональной формы с ребром из N гексагонов КА делится на три равные фигуры, с точностью до одного ряда (рис. 2, а), переходящие друг в друга при повороте на 120° . Центр симметрии выделим как точку отсчета в отдельную «нулевую» фигуру и проиндексируем тройкой $(0, 0, 0)$. Ячейкам ставится в соответствие тройка индексов (i, j, m) , где $m = 1, 2, 3$, а по осям (i, j) каждой трети поля действуют ограничения: $0 \leq j < N$, $1 \leq i < N$. Это позволяет избежать дублирования индексации при соприкосновении фигур.

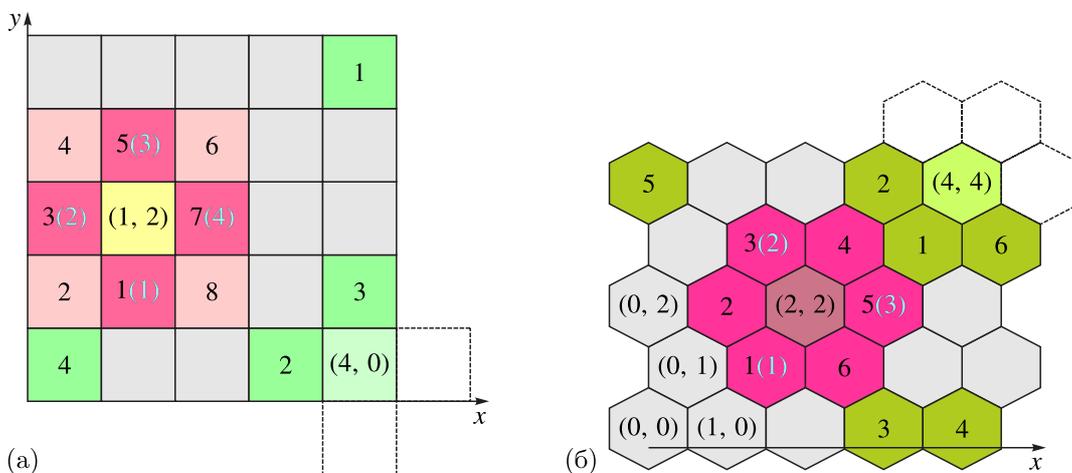


Рис. 1. Глобальные координаты x, y и локальные координаты окрестности для квадратной формы поля КА. (а) для квадратов и $M = 5$; салатовым цветом выделены соседи для граничной ячейки с координатами $(x, y) = (4, 0)$; показаны окрестности Мура и Неймана, а также локальная нумерация в них для ячейки с координатами $(x, y) = (1, 2)$; (б) квадратное поле гексагонов, числа внутри ячеек соседей обозначают номер (порядок) ячейки в окрестности центральной моделирования

При замкнутых границах поля дополнение окрестностей граничных ячеек происходит по следующему правилу:

$$\begin{aligned}
 (N - 1, 0, m) &\rightarrow \{(N - 1, N - 1, 1), (N - 1, N - 1, 2), (N - 1, N - 1, 3)\} \text{ — осевые угловые,} \\
 (N - 1, N - 1, m) &\rightarrow \{(N - 1, 0, 1), (N - 1, 0, 2), (N - 1, 0, 3)\} \text{ — угловые,} \\
 (i, j, m): (i = N - 1, j \neq N - 1, j \neq 0) \vee (i \neq N - 1, i \neq 0, j = N - 1) &\rightarrow \\
 &\rightarrow \{(i, j, m_1), (i, j, m_2)\}: m \neq m_1 \neq m_2 \text{ — на ребре.}
 \end{aligned}
 \tag{7}$$

Здесь $m, m_1, m_2 \in \{1, 2, 3\}$. Реализация последнего варианта показана на рис. 2, а, а второго — на рис. 2, б. Таким способом заданная окрестность легко программируется и удовлетворяет естественному свойству взаимности $A \in \cup(B) \Leftrightarrow B \in \cup(A)$, где \cup — символ окрестности, A, B — индексы ячеек. Алгоритм определения локального номера k для соседа граничной ячейки с координатами (i, j, m) , особенно для гексагонального поля, имеет разветвленную структуру, поэтому его формулировку опускаем.

Выскажем предположение, что в глобальной динамике большую роль играет общее число ячеек $\sim N^2$ (размерный фактор) и связность K , т.е. число соседей. Это ставит гексагональное

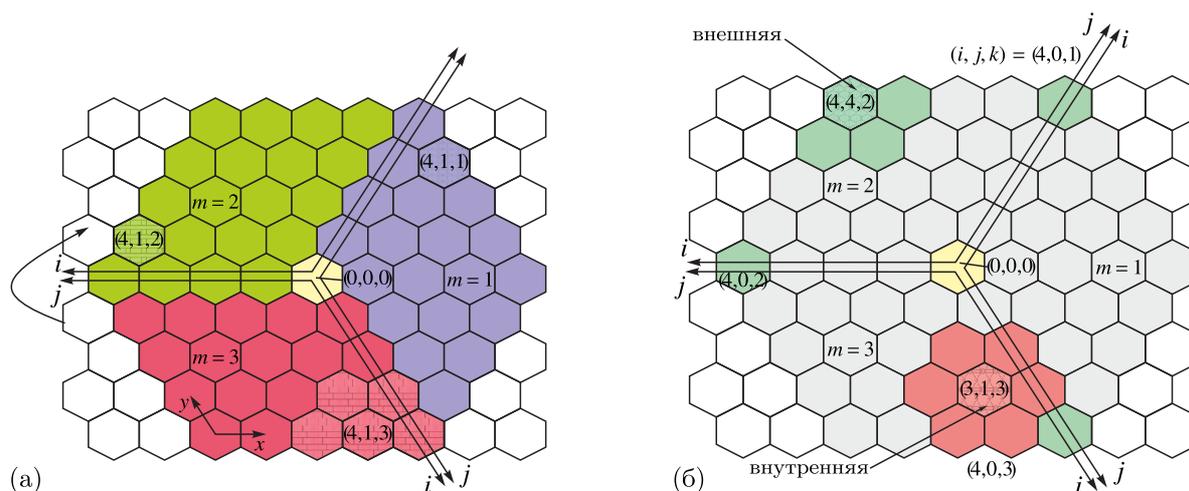


Рис. 2. Гексагональное поле КА на примере $N = 5$ с разметкой координатных осей (а) и демонстрацией (б) формирования окрестностей ячеек. На (б) зеленым цветом выделены ячейки, принадлежащие окрестности граничной (внешней) ячейки $(N - 1, N - 1, 2)$, которая дополняется за счет трех других внешних ячеек по правилу (7). Розовым цветом выделена окрестность внутренней ячейки $(N - 2, N - 4, 3)$, не требующая дополнения. На центральные ячейки окрестностей наложена текстура

поле с шестью соседями между двумя вариантами окрестности квадратного поля: в шаблоне Мура — восемь соседей, а в шаблоне Неймана — четыре; поэтому сравнение именно с этими двумя типами решеток представляется еще более актуальным. Нетрудно получить, что вопрос о соразмерности гексагонального поля ребра (радиуса) N и квадратного поля ребра M сводится к решению диофантова уравнения, аналогичного (и переходящего в пределе) уравнению Пелля: $M^2 - 3N(N - 1) = 1$, причем равенство может выполняться и приближенно. Несколько подходящих пар в формате $(M, N)_d$, где d — дефицит, разность правой и левой частей уравнения: $(6, 4)_{+1}$, $(13, 8)_0$, $(181, 105)_0$, $(1170, 676)_{+1}$, $(16\,296, 9409)_{+1}$.

Предмет исследования определяется режимом работы CMR. Программа работает в двух режимах: элементарном (1) и клеточно-автоматном (2). В элементарном режиме доступны два способа расчета: одиночный, или расчет траектории, и групповой, или мультирасчет (пока не реализовано для КА-режима). При одиночном расчете вычисляется, сохраняется в файл и визуализируется траектория начальной точки $z(0)$, т. е. в элементарном режиме $z(0), z(1), \dots, z(T) \equiv \langle z^t, 0 \leq t \leq T \rangle$ или в клеточно-автоматном режиме $Z(0), Z(1), \dots, Z(T) \equiv \langle Z^t, 0 \leq t \leq T \rangle$. Вводится целое число p , параметр «судьба», отражающее свойства каждой индивидуальной траектории. При групповом расчете, когда меняется только один параметр $q \in \{z(0), \mu, \mu_0\}$, называемый параметром «окна», при фиксированных остальных происходит визуализация двумерной картинке $p(\operatorname{Re} q, \operatorname{Im} q)$. Таким образом, генерируются фрактальные образы множеств Жюлиа и Фату, а также множество Мандельброта.

Вычислительными параметрами являются: T — максимальное число итераций, P — глубина анализа (в итерациях и, очевидно, максимально обнаружимый период меньше P), $\langle \varepsilon, E \rangle$ — уровни отсечки соответственно малого и большого числа. MATLAB-вычисления проводятся с точностью double по стандарту IEEE 754: область значений отрицательных чисел представлена промежутком от $-1,79769 \times 10^{308}$ до $-2,22507 \times 10^{-308}$, положительных чисел — от $2,22507 \times 10^{-308}$ до $1,79769 \times 10^{308}$. Для элементарного расчета алгоритм вычисления p заключается в последовательном совершении следующих шагов.

1. Если $|z(t)| > E$, то расчет обрывается с $p = 0$.
2. Если $|z(t - 1) - z(t)| \leq \varepsilon$, то расчет обрывается с $p = 1$.

Таблица 1. Режимы работы CMR

Способ расчета / Режим работы	Элементарный (один осциллятор, $N = 1$)	Клеточно-автоматный (много осцилляторов, $N > 1$)
Одиночный	Траектория $(\operatorname{Re} z, \operatorname{Im} z)(t)$	Поле цветных ячеек в динамике $C_\alpha(t) = f(z_\alpha(t))$
Групповой (мультирасчет)	$z(0) = \text{const}$: Мандельброт	В разработке
	$\mu, \mu_0 = \text{const}$: Фату/Жюлиа	
Параметр «судьбы» p и его визуализация в мультирасчете	0 — уход в бесконечность (серая палитра) 1 — равновесие (сине-зеленая палитра) $2, 3, \dots$ — периодическая орбита (красно-желтая палитра) Inf — неопределенность, потенциально хаос (фиолетовый цвет)	

3. Для вычисления возможного периода конструируем дополнительный вектор длины $R = \min(p, t)$, равный $\mathbf{r} = (|z(t-1) - z(t)|, |z(t-2) - z(t)|, \dots, |z(t-R) - z(t)|)$. Вычисляем минимум $(r^{\min}, s^{\min}) = \min \mathbf{r}$, $r^{\min} = r(s^{\min})$.

4. Если $(r^{\min} > \varepsilon)$, то расчет продолжается.

5. Если $(r^{\min} < \varepsilon)$, то пусть $(r^{\max}, s^{\max}) = \max(r(1), \dots, r(s^{\min}))$. Если $r^{\max}/r^{\min} > Q$ ($Q \equiv 10$), то завершаем расчет с $p = s^{\min}$. Если же $r^{\max}/r^{\min} < Q$, то завершаем с $p = 1$.

Последняя опция на шаге 5 нужна для учета неподвижных точек типа «фокус». В одиночном элементарном расчете данный алгоритм применяется на каждом шаге и позволяет узнать скорость наступления равновесной, периодической или убегающей динамики. По соображениям быстродействия в групповом элементарном режиме для каждой траектории на каждой итерации вычисляются только 1 и 2 шага алгоритма, и только на последней T -й итерации алгоритм реализуется полностью. В КА-режиме расчет прерывается в том случае, если состояние одной из ячеек превысило порог ($\exists \alpha (|z_\alpha(t)| > E)$). В противном случае исполняются все итерации.

3. Программно-технический базис программы CMR

3.1. Базовые технические параметры

Программа CMR v0.1 написана на языке программирования MATLAB, и системным требованием для ее корректной работы является наличие на устройстве пользователя установленного пакета MATLAB R2019b с расширением для поддержки параллельных вычислений Parallel Computing Toolbox. Файловая структура программы — 24 М-файла исходных кодов и 5 файлов графических элементов формата fig. Точка входа находится в файле CA.m. Объем дисковой памяти, занимаемый всеми файлами, — 697 Кб. Самая ранняя совместимая с CMR версия MATLAB — релиз R2017b, последняя — R2019b.

При наличии требуемой платформы и расширения запуск программы производится из консоли среды разработки MATLAB. Папку с программой можно не устанавливать в качестве текущей. Если на устройстве пользователя нет среды MATLAB или ее версия несовместима с программой, установка собранной в исполняемый файл программы выполняется при помощи дистрибутива. Дистрибутив программы [Github-репозиторий программы. . . , 2021] проверяет наличие интерпретатора и расширения MATLAB на компьютере пользователя и скачивает их из интернета либо распаковывает из своего внутреннего архива. Соответственно, существует два способа подготовки программы к работе:

- 1) web-установщик, который при запуске дистрибутива программы скачивает из интернета и устанавливает интерпретатор MATLAB и расширение Parallel Computing Toolbox; данный установщик занимает около 2 Мб памяти и требует подключения к сети интернет;

- 2) offline-установщик, который при запуске дистрибутива программы устанавливает на компьютере встроенный в него интерпретатор и расширение MATLAB; дистрибутив не требует подключения к Сети, однако имеет вес около 1 Гб.

Общий вес файлов установленной программы составляет 197 Мб, интерпретатора — 2,02 Гб.

При помощи встроенного в MATLAB модуля профилирования пользовательских подпрограмм MATLAB Profiler была проведена оценка времени типового моделирования. Расчеты производились на компьютере со следующими характеристиками: процессор — AMD Ryzen 5 3500U (число ядер — 4, число потоков — 8, базовая тактовая частота — 2,1 ГГц); ОЗУ — 6 Гб DDR4; твердотельный накопитель — WDC PC SN530 (скорость чтения — 2244 Мб/с, записи — 899 Мб/с). В таблице 2 представлены описание проводимых расчетов и время, прошедшее с момента нажатия на кнопку «Смоделировать» до сохранения результатов на диск, производимого после визуализации. В анализируемых расчетах пороги бесконечности и сходимости составляли 10^{15} и 10^{-5} , соответственно.

Таблица 2. Среднее время моделирования типовых расчетов

Режим	Описание проведенного расчета	Затраченное время в секундах
Групповой элементарный	Групповой расчет по параметру μ_0 для отображения $g: z \rightarrow (\mu + \mu_0) \times \exp(i \cdot z)$ с числовыми параметрами $\mu_{0_{centr}} = \mu = z_0 = 0$, $(\text{Re}, \text{Im})\mu_0 \in \in [-5; 5]^2$, $T = 1000$, $P = 200$	664
Одиночный КА-расчет	КА-расчет с ЛФП автомата в виде итерированной функции $z_\alpha(t+1) = \mu + \mu_0 \left \frac{1}{K} \sum_{k=1}^K (-1)^j z_k(t) \right \cdot \exp(i \cdot z_\alpha(t))$ и со следующими основными настройками: вид поля — гексагональное, шаблон окрестности — Мура, тип границ — замкнутые, ориентация гексагонов — вертикальная, $N = 5$. Начальную конфигурацию ячеек поля задать равномерным центральным распределением с параметрами $z_0 = a = 0$, $b = c = 1$. Веса ячеек в локальной окрестности: $\forall k = 1, 2, \dots, 6: \mu_k = 1$. Параметры отображения: $\mu = 1$, $\mu_0 = 0,25i$. Пороги бесконечности и сходимости — 10^{15} и 10^{-5} соответственно, $T = 100$	284
Одиночный элементарный	Элементарный одиночный расчет при отображении $g: z \rightarrow \mu z + \mu_0 z^{-1}$, начальном значении аргумента $z_0 = 1 + 1i$ и параметрах $\mu = 1$, $\mu_0 = 0,00012 + 0,0132i$. Настройки моделирования: $T = 10\,000$, $P = 500$. Расчет окончен на итерации 3252. «Судьба» — период 238	183

Примечание. Время расчета сильно зависит от скорости определения «судьбы», поэтому приведенные в таблице данные носят преимущественно ознакомительный характер.

3.2. Структура пользовательского интерфейса и маршрут моделирования

Окно программы структурно подразделяется на восемь основных областей, обозначенных цифрами на рис. 3. Работа в программе SMR начинается с последовательного взаимодействия с панелями 1–4, которые включают в себя элементы ввода данных для определения итерированного объекта моделирования, что соответствует обобщенной последовательности действий пользователя.

1. Панель «Основные параметры решетки КА» содержит группы селекторов, отвечающих за выбор главных характеристик итерированного объекта. В случае КА здесь задаются главные настройки автомата: тип границ и форма поля, вид ячеек, шаблон окрестности ячейки.

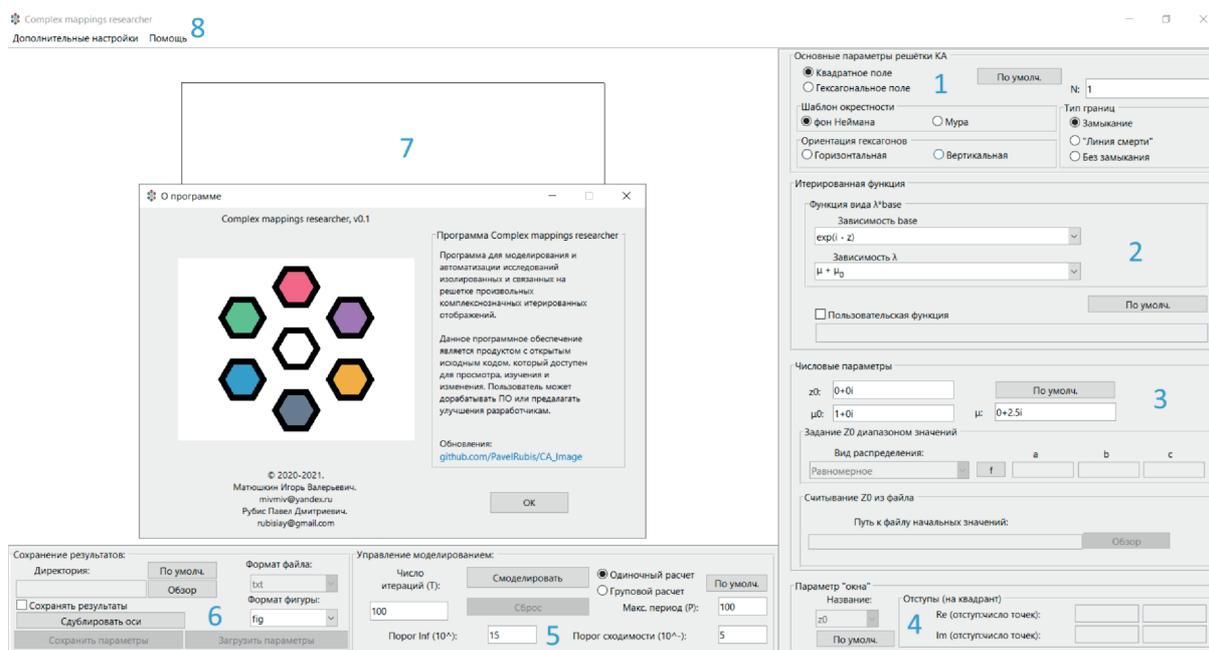


Рис. 3. Главное окно программы CMR. Цифрами выделены основные функциональные блоки (панели) пользовательского интерфейса

В текстовое поле N вводится целочисленное значение ребра КА или единица для одиночного отображения.

2. Панель «Итерированная функция» позволяет задать закон итераций для моделируемого объекта. Функция отображения вводится в текстовом поле после установки соответствующего флажка. При элементарных расчетах интерпретация введенной функции производится по следующим правилам: аргумент отображения обозначается символом z ; в выражении могут присутствовать параметры μ, μ_0 , обозначаемые как mi и $mi0$ соответственно; знаки умножения, деления, возведения в степень и разделения целой и дробной частей числа обозначаются как «*», «/», «^», «.»; аргументы элементарных математических функций должны стоять в скобках ($abs(z - mi)$); $\exp(i * z)$; $\sin(z + mi0)$, а сами математические функции должны быть вызваны в формате, распознаваемом интерпретатором MATLAB. Распознавание программой пользовательской функции при КА-моделировании отличается только тем, что в выражении также могут присутствовать значения соседних ячеек в окрестности $-z_k$ ($k = 1, 2, \dots, K$), а также весовые коэффициенты центральной ячейки и соседей $-\mu_c, \mu_k$ ($k = 1, 2, \dots, K$), задаваемые в соответствующем окне (рис. 4). В выражении итерированной функции данные параметры обозначаются как $z0, mic, zk, mik$ ($k = 1, 2, \dots, K$). Пользователю также доступны встроенные отображения вида (2). Их задание происходит посредством выбора зависимостей из двух раскрывающихся списков в группе «Функция вида $\lambda * base$ ».

3. Панель «Числовые параметры» позволяет задать коэффициенты μ, μ_0 и начальное состояние итерированного объекта. В элементарном одиночном и КА-расчете μ, μ_0 являются внешними параметрами отображения, z_0 — начальным значением точки или репером распределения $Z(0)$. Эти параметры служат константами в функциях семейств отображений мультирасчета, когда два параметра фиксируются, а третий параметр, «окно», задает центр множества значений изменяющегося параметра (см. панель 4 на рис. 3).

Ниже текстовых полей расположены субпанели для задания начальной конфигурации автомата в КА-расчете. В селекторе «Задание Z0 диапазоном значений» исходные состояния ячеек устанавливаются через настройку генерирующего распределения. Также можно открыть окно с формулой распределения, нажав на кнопку f справа от селектора. Любая выбранная функция зависит от координат ячейки (x, y) и содержит параметры a, b, c , задаваемые в текстовых полях правой половины панели 3.

4. Панель «Параметр «окна» используется только при мультирасчете для создания множества изменяемого параметра. При помощи селектора слева выбирается сам параметр, а в группе справа задаются отступы от центрального значения и число точек в квадранте как для реальных компонентов элементов множества, так и для мнимых.

5. Панель управления моделированием. После установки итерированного объекта нужно задать параметры точности расчета (пороги бесконечности E и сходимости ϵ , а также глубину T и максимальный период P). Это обеспечивается при помощи соответствующих текстовых полей панели 5 (рис. 3). На панели также размещены кнопки «Смоделировать» и «Сброс», управляющие жизненным циклом объекта моделирования. Итерированный объект создается в оперативной памяти только после нажатия «Смоделировать». В одиночном элементарном и КА-расчетах этот же объект эволюционирует при каждом последующем нажатии данной кнопки. То есть в данных режимах моделирование может производиться в несколько этапов; например, пользователь может промоделировать 15 итераций работы КА, затем изменить параметр T на 20 и снова нажать на «Смоделировать». В итоге будет произведено 35 итераций. Такое же поведение итерированного объекта реализовано и при расчете траектории точки. Кнопка «Сброс» удаляет текущий итерированный объект из оперативной памяти и делает доступными GUI-элементы для создания нового. Рекомендуется не начинать моделирование до настройки сохранения результатов.

6. Панель сохранения результатов моделирования отвечает за выбор формата сохраняемых после моделирования данных, а также позволяет выбрать директорию файловой системы компьютера для их записи. Флажок «Сохранять результаты» определяет необходимость записи выходных данных расчета. При установленном флажке и отсутствии выбранной директории результаты моделирования будут сохранены в папку с программой. На панели также присутствует кнопка «Сдублировать оси», при нажатии на которую создается MATLAB-фигура, содержащая текущие оси главного окна, что обеспечивает удобство редактирования.

7. Окно визуализации изначально представляет собой белый прямоугольник пустых осей. После окончания моделирования на нем визуализируются результаты. Смена визуализации при необходимости производится в меню программы.

8. Меню программы. До и после проведения одиночного элементарного или КА-расчетов в меню программы можно задать параметры визуализации результатов моделирования, выбрав в раскрывающемся списке «Дополнительные настройки» соответствующие элементы «Визуализация траектории точки» или «Визуализация поля КА» (в зависимости от вида моделирования). Выбор одной из этих опций приводит к открытию диалогового окна настройки графика выходных данных (рис. 4, а). При КА-моделировании в меню программы для пользователя также важен элемент «Установка весов ячеек КА». Этот пункт меню вызывает окно установки весов соседних ячеек окрестности клетки КА (рис. 4, б).

3.3. Архитектура программы

Структура классов вычислительного блока программы была реализована с учетом математической специфики моделируемых объектов и с полным соблюдением принципов объектно-ориентированного программирования (SOLID) [Hall, 2017]. На рис. 5 представлено строение математического ядра CMR в виде UML-диаграммы классов.

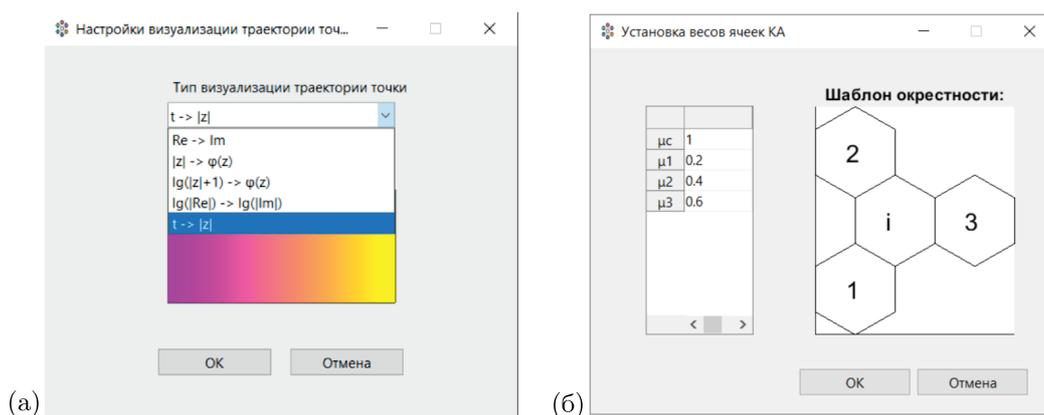


Рис. 4. Окна настройки визуализации результатов одиночного элементарного моделирования (а) и установки весов ячеек в окрестности при КА-расчете (б)

Все доступные в программе виды расчета (объекты моделирования) наследуют и реализуют абстрактный класс *IteratedObject*. Он определяет итерированную функцию *IteratedFunc*, массив числовых параметров для итерированной функции *funcParams* и абстрактные методы *Iteration*, *Initialization* и *BeforeModeling*, требующие реализации в дочерних классах. Методы принимают на вход экземпляр текущего класса и возвращают его же.

Класс *IteratedPoint* описывает объект точки (одиночный расчет с $N = 1$), содержит свойства *InitState*, *statePath*, *fate*, *lastIterNum*, обозначающие соответственно: начальное состояние точки, массив-траекторию, «судьбу», итерацию, на которой была определена «судьба». Внутри класса *IteratedPoint* определяется статический метод анализа «судьбы» точки *CheckConvergence*.

Сущность *IteratedMatrix* представляет собой объект для множественного моделирования. Помимо свойств и методов родителя *IteratedObject*, класс реализует свойства *windowOfValues*, *windowParam*, *pointsFates*, *pointsSteps*, означающие соответственно: матрицу начальных аргументов изолированных отображений, структуру с названием варьируемого параметра и его центральным значением, матрицу судеб и матрицу итераций идентификации судьбы. Элементы последних двух матриц ставятся в соответствие элементам из *windowOfValues*. Класс *IteratedMatrix* зависит от класса *IteratedPoint*, так как использует статический метод *CheckConvergence*.

Класс *CA* представляет собой объект автомата для КА-расчета. Он включает в себя объект окрестности *NeighbourHood* через связь-композицию, ячейки автомата (объекты *CAcell*) — через связь типа «один ко многим». В классе определяются интуитивно понятные свойства N , *caCellsWeight* — ребро поля и массив весов ячеек в окрестности клетки КА соответственно; также реализуются методы для генерации гексагонального и квадратного полей ячеек (*GenerateHexField*, *GenerateSquareField*), методы для установки начальной конфигурации поля клеток *InitCellsWithRandRange* (при помощи значений выбранной функции) и *InitCellsWithFile* (путем чтения файла).

Абстрактный класс *CAcell* описывает базовый класс объекта ячейки. Класс имеет свойство *CAIndexes*, реализуемое в дочерних классах для однозначного определения ячейки на поле КА. Свойства z_0 , $zPath$, *isExternal*, *currNeighbors* и *renderColor* обозначают соответственно начальное состояние ячейки, массив-траекторию, массив ячеек-соседей, цвет для визуализации на поле. Кроме того, *CAcell* определяет пять абстрактных методов, реализуемых в дочерних классах: *Render* для отрисовки ячейки КА на поле; *GetMooreNeighboursIndexes* и *GetNeumannNeighboursIndexes*, которые возвращают для текущей клетки индексы ячеек-соседей в массиве ячеек всего поля в зависимости от шаблона окрестно-

сти; *GetAllMooreNeighborsPlaces* и *GetAllNeumannNeighborsPlaces*, которые возвращают индексы ячеек-соседей в локальной окрестности текущей ячейки также в зависимости от шаблона.

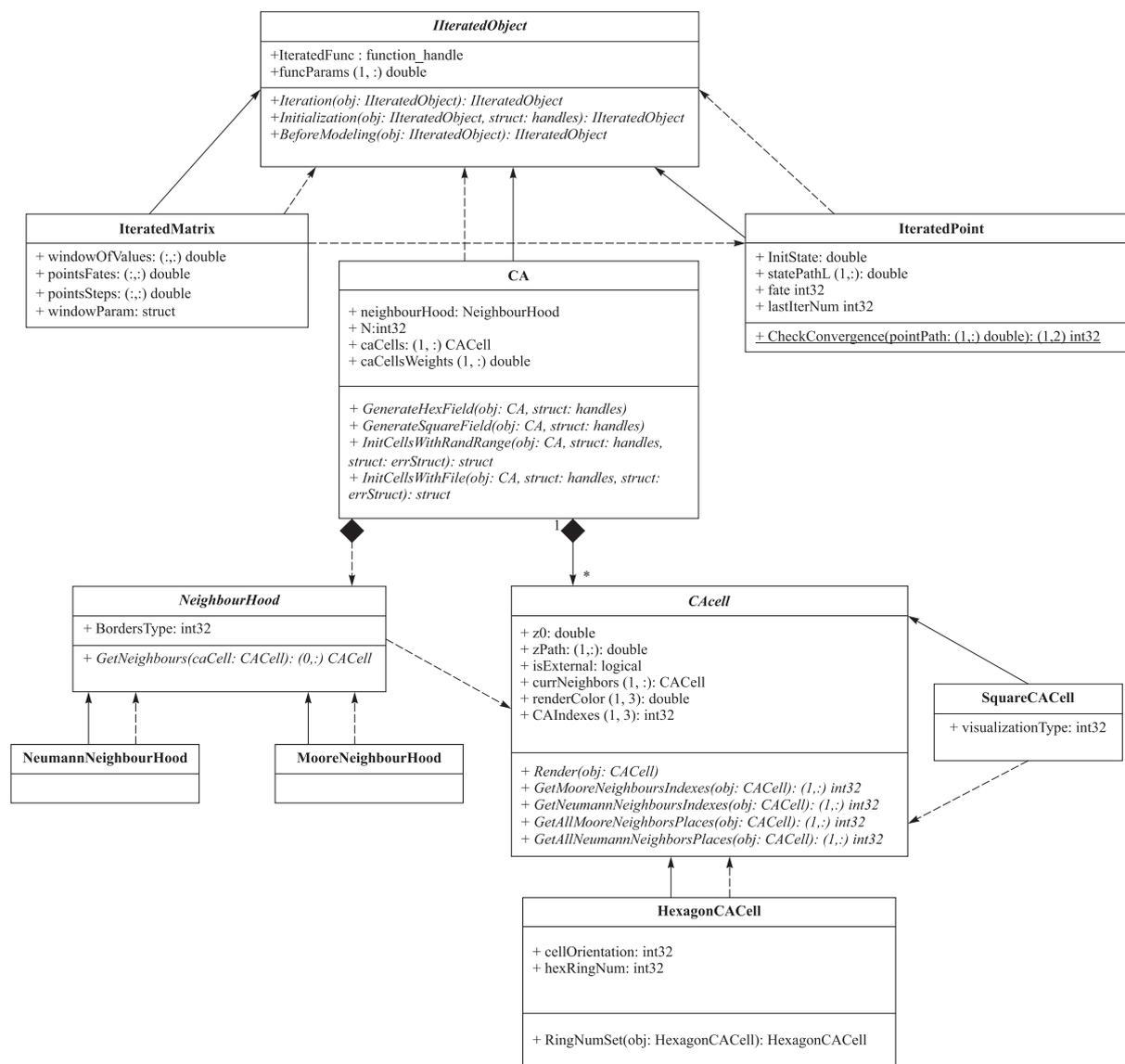


Рис. 5. UML-диаграмма классов математического ядра Complex mappings researcher

Классы *HexagonCACell* и *SquareCACell* представляют гексагональную и квадратную ячейки соответственно. Оба наследуют и реализуют класс *CACell*; первый из классов дополнительно содержит свойство *hexRingNum* и set-метод *RingNumSet*, которые нужны для определения номера кольца из гексагонов, в котором находится ячейка на поле КА. Это необходимо для генерации исходной конфигурации КА при помощи центрального равномерного распределения. Свойства *cellOrientation* и *visualizationType* в классах *HexagonCACell* и *SquareCACell* соответственно отвечают за способ визуализации ячейки при отрисовке.

Абстрактный класс *NeighbourHood* определяет свойство *BordersType*, обозначающее тип границ поля КА. Класс определяет абстрактный метод *GetNeighbours* для получения соседей переданной ячейки. Метод реализуется дочерними классами *NeumannNeighbourHood* и *MooreNeighbourHood*.

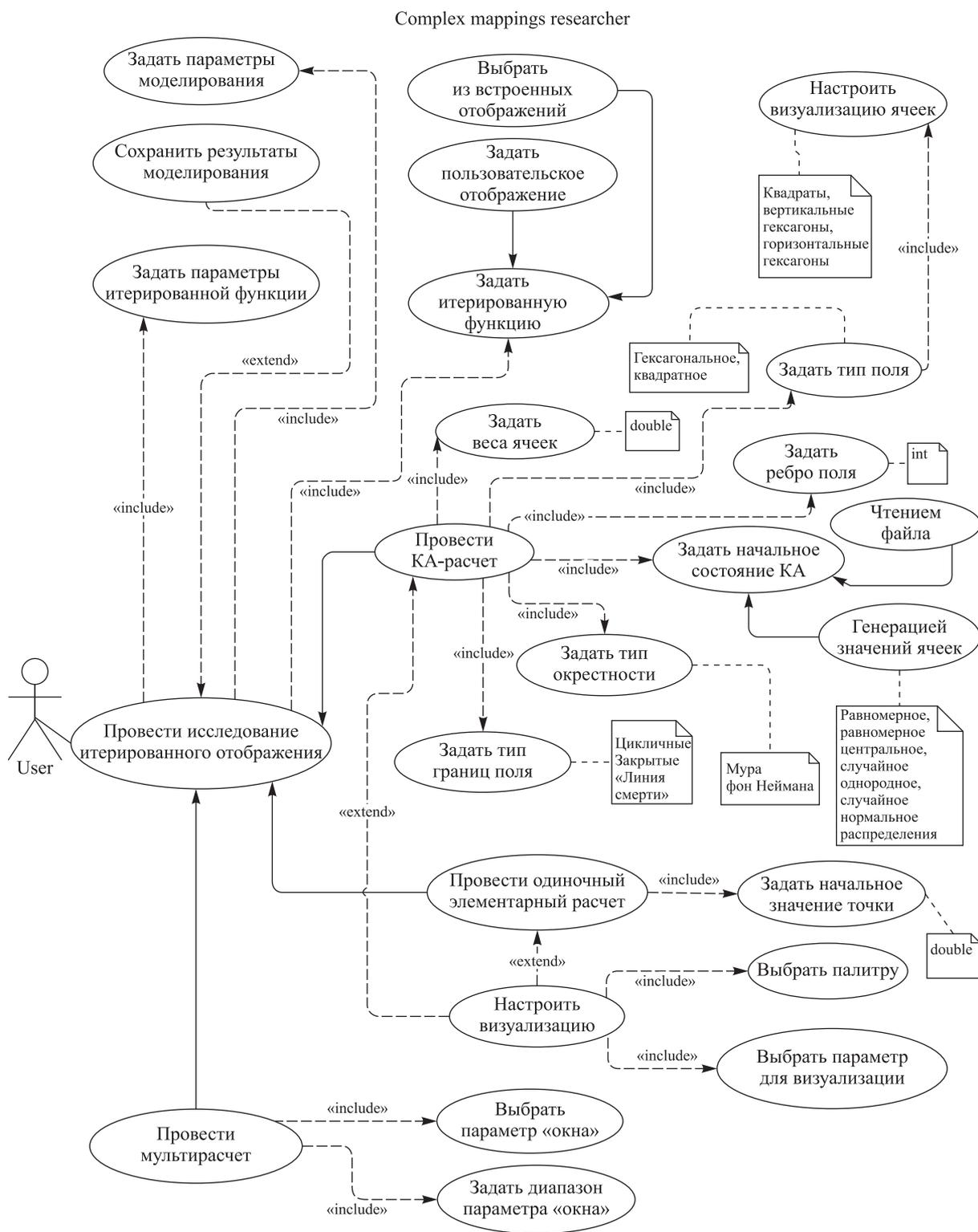


Рис. 6. Диаграмма вариантов использования (use case) программы CMR

При разработке программы CMR была учтена возможность изменения визуализации моделирования как перед, так и после расчетов. Поэтому параметры визуализации необязательно

выбирать перед моделированием, так как для них имеются значения по умолчанию, что отражено на диаграмме вариантов использования программы (рис. 6). На ней видно, что настройка визуализации не включена в прецеденты одиночного и КА-расчетов, а только расширяет их, не являясь обязательной.

3.4. Особенности программирования в MATLAB

Классы в большинстве высокоуровневых языков программирования по умолчанию являются ссылочными объектами. Классы в MATLAB по умолчанию есть типы значений, конкретный класс становится ссылочным типом данных только после наследования от специального класса *handle*. В платформе отсутствуют интерфейсы, однако доступно множественное наследование, что делает возможной ООП-архитектуру.

Для ускорения расчета в клеточно-автоматном и групповом режимах было реализовано распределение вычислений по ядрам центрального процессора компьютера пользователя с помощью расширения среды MATLAB Parallel Computing Toolbox и замены в М-файлах ключевого слова *for* на *parfor*. Не отрицая значительное ускорение расчетов при использовании параллелизма, стоит, однако, отметить, что для реализации многопоточных вычислений MATLAB создает специальный пул потоков Parallel Pool, что занимает некоторое время перед непосредственным стартом моделирования.

Графический интерфейс программы SMR сделан при помощи встроенного в MATLAB редактора GUIDE, позволяющего быстро и эффективно создавать графические пользовательские интерфейсы через интерактивное взаимодействие с макетом конкретного окна. Кроме того, реализация программы усложнилась бы без использования встроенного в MATLAB типа *cell*. Этот тип данных является контейнером, в который может быть упакован объект любого другого типа. Среда MATLAB предоставляет широкий набор функций для работы с переменными типа *cell*. С его помощью были реализованы методы и функции с варьируемым числом входных и выходных аргументов.

4. Демонстрации

Демонстрация 1. Одиночный элементарный расчет

Формулировка. В два этапа провести моделирование орбиты точки со следующими настройками: отображение $g: z \rightarrow z^{\mu_0} \cdot \exp(i \cdot z \cdot \mu_0)$, $z_0 = 1 + 1i$, $\mu_0 = 0,5 + 0,5i$. Настройки моделирования на двух этапах: $T_1 = P_1 = 40$, $T_2 = P_2 = 30$, пороги бесконечности и сходимости — 10^{15} и 10^{-5} соответственно.

Перед началом моделирования необходимо убедиться, что на панели 5 (рис. 3) выбран селектор «Одиночный расчет». После нажатия на него становятся доступны только элементы графического интерфейса, отвечающие за одиночное моделирование (клеточно-автоматный и элементарный расчеты). Далее в текстовое поле N панели 1 вводится единица. Требуемая функция отображения по структуре сходна с библиотечными, но выходит за их пределы, поэтому идем общим путем. В панели 2 ставится флажок «Пользовательская функция» и вводится выражение $z^{\mu_0} \cdot \exp(i \cdot z \cdot \mu_0)$ без каких-либо знаков поэлементности. В панели 3 задаются z_0 и μ_0 . Так как в выбранном отображении отсутствует параметр μ , то соответствующее текстовое поле оставляется пустым. Предпоследним шагом перед выполнением расчета является настройка сохранения результатов в панели 6. В завершение настройки производится ввод параметров моделирования и нажимается кнопка «Смоделировать».

Названия выходных файлов данных и их визуализации будут одинаковы — «Modeling-<День>-<Месяц>-<Год> <Часы>-<Минуты>-<Секунды>-N-1-Path». Так как исследование итерированной функции проводится в два этапа, то после отрисовки результатов на панели 7 T_1

заменяется на T_2 и кнопка моделирования нажимается снова. Между этапами моделирования и после завершения всего расчета может быть изменена визуализация. Так, в данной демонстрации после первого этапа расчета была произведена смена палитры и зависимости для отрисовки (рис. 7, а, б). Также стоит отметить, что на осях 7 всегда визуализируется часть результатов, соответствующая последнему этапу моделирования.

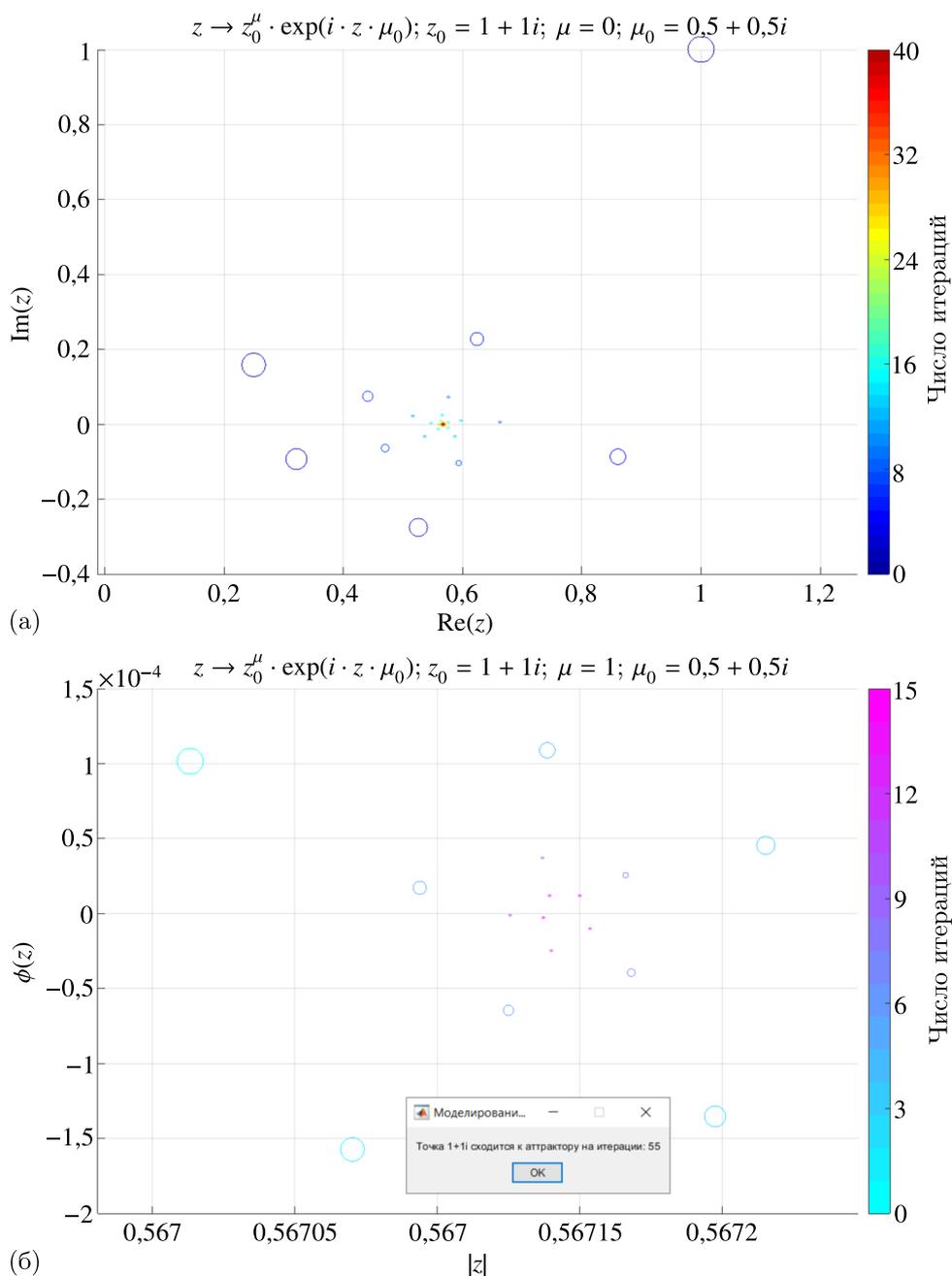


Рис. 7. Визуализации этапов одиночного элементарного расчета из демонстрации 1. На обоих рисунках отрисованы реальная и мнимая части траектории точки $z_0 = 1 + 1i$. Модальное окно на рисунке (б) уведомляет о нахождении судьбы точки и, соответственно, о прекращении моделирования с заданными параметрами точности

При сохранении результатов одиночного элементарного расчета в выходной файл формата .txt, помимо характеристик итерированного объекта, записывается значение «судьбы» точки, а также вся траектория точки в виде массивов реальных и мнимых составляющих. Если расчет многоэтапный, файл дополняется на каждом этапе. Между этапами моделирования пользователю доступны пункты меню программы для настройки визуализации, а также элементы панели 5, поэтому для каждого этапа расчета могут быть заново заданы параметры T , P , E , ε .

Демонстрация 2. Групповой элементарный расчет (мультирасчет)

Формулировка. Получить множества Фату, Жюлиа (по окну параметра μ_0) и Мандельброта (по окну z_0) для отображения $g: z \rightarrow (\mu + (\mu_0 \cdot |z - z^*|)) \cdot \exp(iz)$ при следующих величинах: $z_0 = 0$, $\mu = \mu_0 = 1i$, $T = 1000$, $P = 100$, отступы по обеим осям окна — 5, количество точек на квадрант — 500.

Для начала новой демонстрации предыдущий итерированный объект удаляется путем нажатия кнопки «Сброс» в панели 5 (рис. 3). Далее там же выбирается групповой режим расчета путем нажатия на соответствующий переключатель. После этого в группе «Функция вида $\lambda \cdot \text{base}$ » панели 2 осуществляется выбор двух составляющих библиотечной функции в раскрывающихся списках. В панели 4 через раскрывающийся список выбирается варьируемый параметр μ_0 , затем вводятся число точек на реальной и мнимой сторонах квадранта и отступы по обеим осям от центрального значения параметра окна (рис. 8).

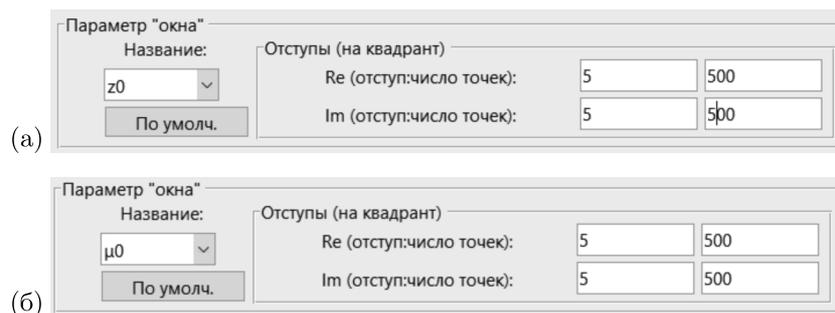


Рис. 8. Панель 4 перед первым (а) и вторым (б) моделированиями из демонстрации 2

Последними задаются параметры моделирования. Визуализации расчетов представлены на рис. 9.

При сохранении результатов мультирасчета в выходной файл формата .txt, помимо характеристик итерированного объекта, записываются последние значения всех итерированных функций семейства и соответствующие им «судьбы».

Демонстрация 3. Одиночный КА-расчет

Формулировка. В два этапа провести КА-расчет с ЛФП автомата в виде итерированной функции $z_\alpha(t+1) = \mu_0 + \sum_{k=1}^K \mu_k z_k \cdot \exp(i \cdot z_\alpha(t))$ и со следующими основными настройками: вид поля — гексагональное, шаблон окрестности — фон Неймана, тип границ — замкнутые, ориентация гексагонов — вертикальная, $N = 5$. Начальную конфигурацию ячеек поля задать равномерным центральным распределением с параметрами $z_0 = a = 0$, $b = c = 1$. Веса ячеек в локальной окрестности: $\mu_c = 1$, $\mu_1 = 0,2$, $\mu_2 = 0,4$, $\mu_3 = 0,8$. Параметры отображения: $\mu = 1$, $\mu_0 = 0,25i$, $T_1 = T_2 = 5$.

В начале необходимо очистить предыдущее моделирование (панель 5, кнопка «Сброс» на рис. 3). На этой же панели после нажатия на переключатель «Одиночный расчет» пользователю становятся доступны элементы графического интерфейса для задания описания клеточного

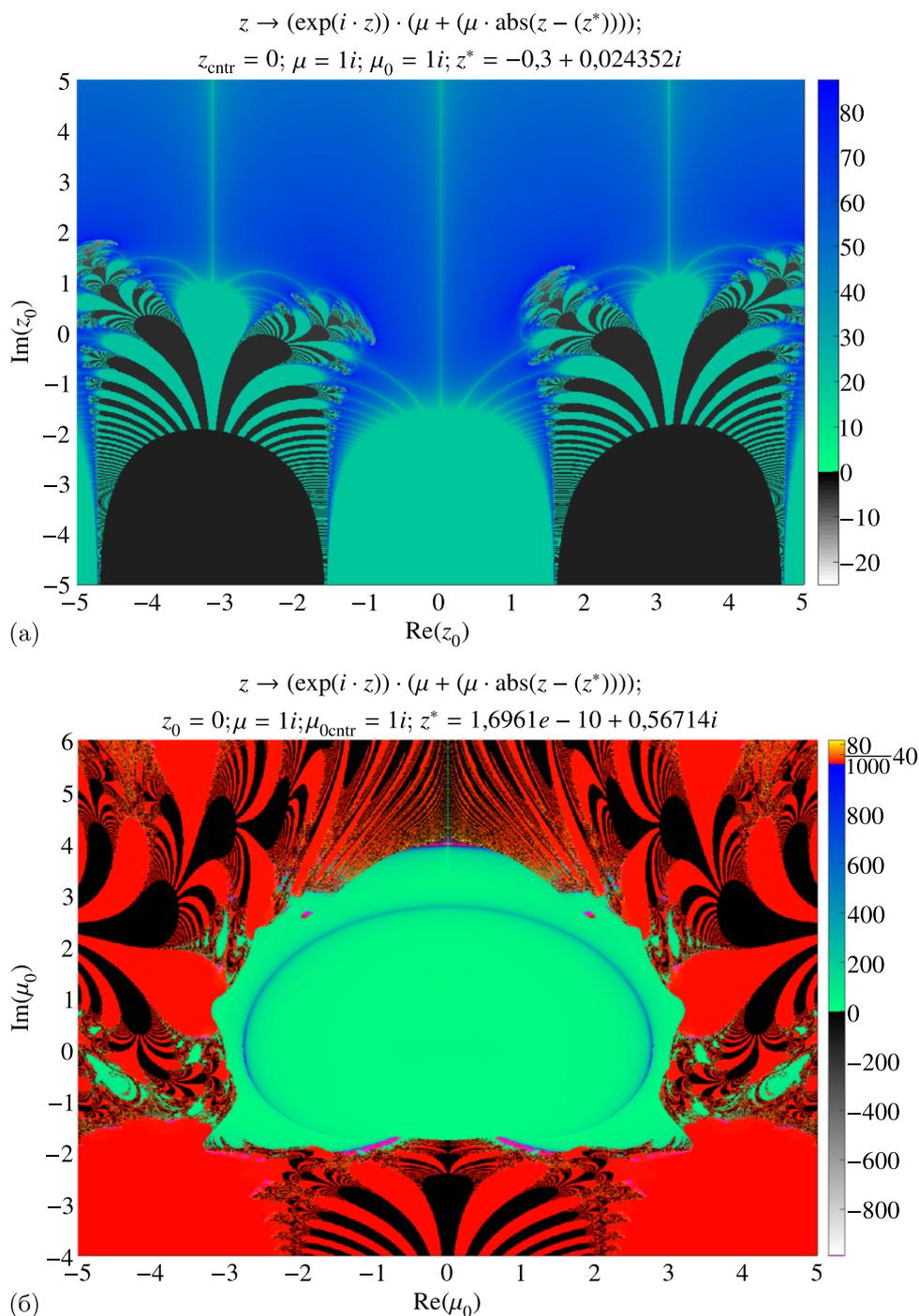


Рис. 9. Визуализации множеств Жюлиа (а) Мандельброта (б) как результаты групповых расчетов по параметрам z_0 и μ_0 ; отчетливо визуализируются букеты Кантора

автомата. Основные настройки КА, упомянутые в формулировке, устанавливаются при помощи селекторов и текстового поля в панели 1. Задание итерированной функции, а также числовых параметров z_0, μ, μ_0 описано в предыдущих демонстрациях. Для генерации исходного состояния поля автомата требуется выбрать тип распределения в раскрывающемся списке субпанели «Задание Z0 диапазоном значений». Параметры a, b, c генерирующей функции вводятся в текстовые

поля рядом. Для задания весов соседних ячеек в меню программы нужно выбрать элемент «Установка весов ячеек КА» пункта «Дополнительные настройки». После чего откроется модальное окно, в котором, в зависимости от установленных ранее основных параметров КА, будет изображен шаблон окрестности ячейки автомата с пронумерованными соседями. Ввод весов производится в таблице, расположенной рядом с макетом окрестности (рис. 4, б). Последними задаются параметры моделирования. Как и в одиночном элементарном расчете, между этапами моделирования пользователю доступны пункты меню программы для настройки визуализации, а также элементы панели 5 (рис. 3). Для каждого этапа расчета могут быть заново заданы параметры T , P , E , ε .

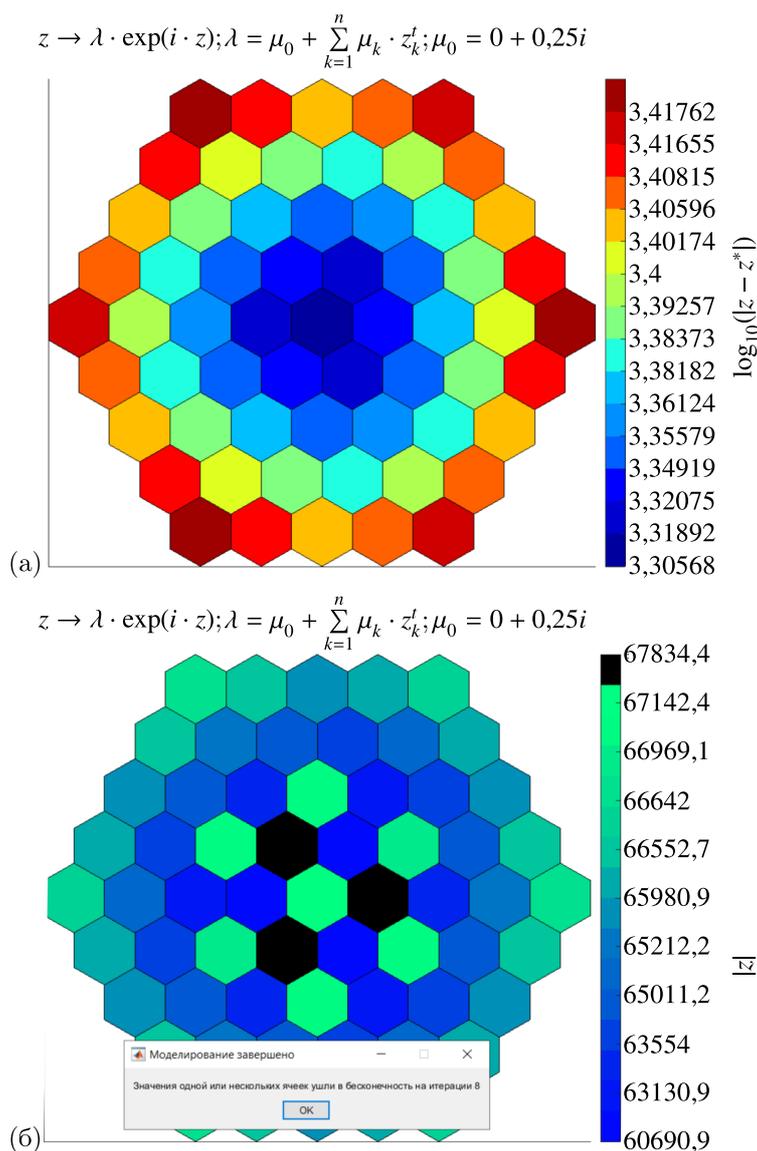


Рис. 10. Поле ячеек автомата на итерации $t = 5$ (а) и после завершения моделирования на итерации $t = 9$ (б). Цвета ячеек соответствуют значениям $\log(|z - z^*|)$ (а) и $|z|$ (б)

После отрисовки поле КА доступно для взаимодействия: по щелчку мыши координаты и состояние ячейки отображаются в строке под визуализацией. Расчет с заданным порогом бесконечности прекращается, если состояние хотя бы одной из ячеек автомата превысило это зна-

чение. Как и при одиночном элементарном расчете, пользователю сообщается о его окончании в соответствующем модальном окне. Визуализации расчетов представлены на рис. 10, *а, б*.

При сохранении результатов КА-моделирования в выходной файл формата .txt, помимо характеристик итерированного объекта, записываются состояния ячеек поля автомата перед текущим этапом расчета и все конфигурации данного этапа. Для каждого этапа моделирования создается новый файл результатов.

Демонстрация 4. Исследование отображений $\mu z + \mu_0 z^{-1}$, $\mu_0 + z^{\mu}$

Рабочая версия программы позволила нам исследовать [Матюшкин, 2016; Matyushkin, Zapletina, 2018; Матюшкин, Заплетина, 2018] голоморфную динамику отображения с базой $\exp(iz)$ и модификатором λ в виде линейной функции: наиболее интересные результаты были получены для линейного модификатора, содержащего знак модуля. Также мы исследовали дробно-линейное отображение с двумя параметрами (вида $\mu z + \mu_0 z^{-1}$), у которого имеется особенность в точке $z = 0$, причем в нее можно попасть из бесконечного числа точек, например $z_{-1} = \pm i \sqrt{\mu \cdot \mu_0}$. В литературе разбирались случаи более высоких порядков: $z^n + \lambda z^{-n}$, $n \geq 2$ [Alexander, Devaney, 2015]. Как известно, для топологии множеств Фату и Жюлиа характерны зависимости от семейства исходной функции. Так, для полиномов были получены фигуры типа «пряничного человечка» классического множества Мандельброта, для экспонент или гиперболических функций вида $\text{sh}(z)$ — букеты Кантора, а для дробно-линейных — ячеистая структура, где встречаются кольца Эрмана или диски Зигеля.

С точки зрения экспериментальной математики исследование отображения $\mu z + \mu_0 z^{-1}$ затруднительно в силу «поворота на иррациональный угол», вследствие чего наблюдается хаотическая динамика, т. е. при одиночном элементарном расчете задействованы все T итераций, причем $T \gg 1000$ (случай тяжелой сходимости). По той же причине, в силу ограниченности величины T , при групповом режиме большая часть точек не успевает сойтись к какому-либо периоду (окрашены в розовый цвет на рис. 11, *а*). Очевидно также, что возрастает роль погрешности вычислений на долгом периоде. Требование устойчивости разделяет комплексную плоскость на «лоскуты» устойчивых и неустойчивых областей, размеры и формы которых зависят от пары $\langle \mu, \mu_0 \rangle$.

На рис. 11, кроме двух больших множеств периодических точек и сходящихся к равновесию, заметны структуры из точек с малым периодом — «пряничные человечки» классического множества Мандельброта. Также диагностируются точки с большей, чем 2, величиной периода. На рис. 12 показаны визуализации поля КА с ЛФП в виде отображения $z_{\alpha}(t+1) = \mu_0 + \sum_{k=1}^K \mu_k z_k \cdot (z + i \cdot z_{\alpha}(t)^{-1})$ на итерациях $T_1 = 3$ и $T_2 = 5$.

На рис. 13 показана визуализация результатов работы программы для отображения $g: z \rightarrow \mu_0 + z^{\mu}$. Здесь параметр μ расположен нестандартно, благодаря чему исследуется влияние степени отображения на судьбу отдельной точки. Взаимопроникновение областей короткопериодических режимов и режима убегания показывает непривычный зубчатый рельеф (рис. 14), и еще более интересна топология длиннопериодических и хаотических режимов на рис. 15.

Заключение

Представленная программа SMR позволяет строить и визуализировать одиночные комплекснозначные отображения и их коллективы как линейно связанные на квадратной или гексагональной решетке. Возможности исследования итерированной динамики коллективов комплекснозначных отображений пока находятся в зачаточном состоянии. Например, не ясны обобщения множеств Жюлиа, Фату и Мандельброта на модель клеточного автомата, ощущается недостаток теоретических конструкций, каким-то образом агрегирующих такую динамику. Вместе с тем

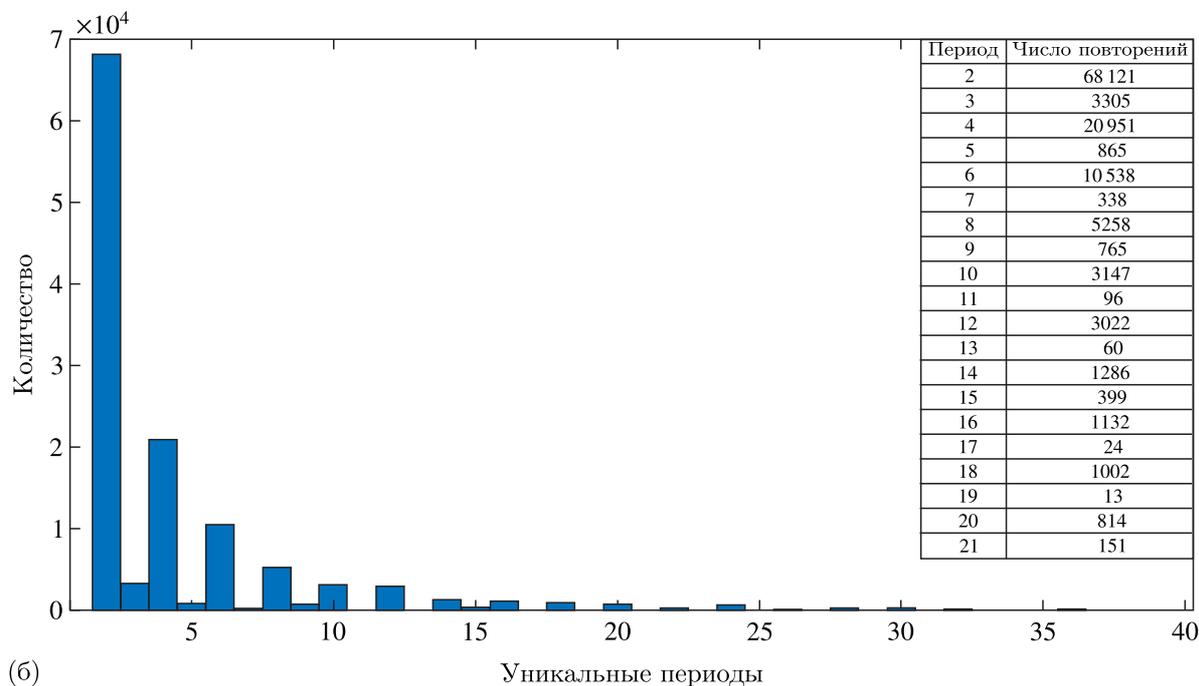
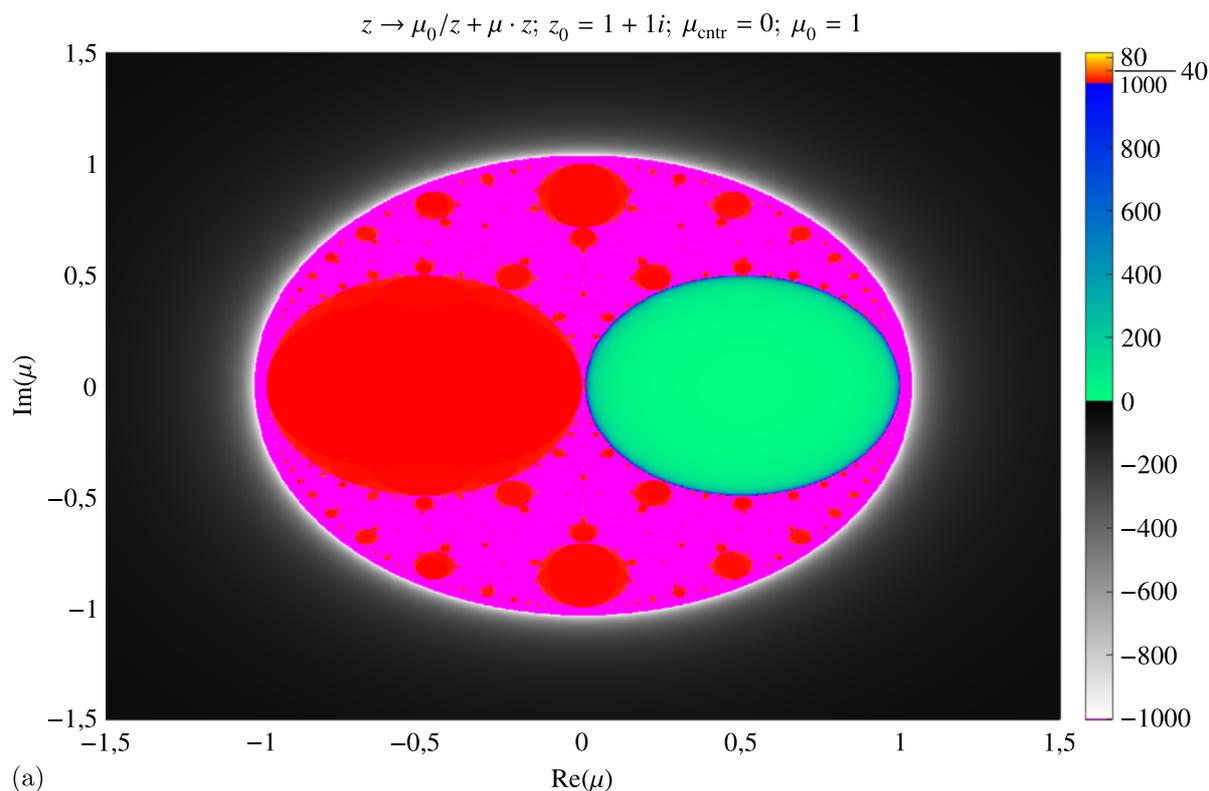


Рис. 11. Множество Мандельброта (а) (серые тона) по параметру μ для отображения $g: z \rightarrow \mu z + \mu_0 z^{-1}$ с параметрами $\mu_{\text{cntr}} = 0, \mu_0 = 1, (\text{Re}, \text{Im})\mu_0 \in [-1,5; 1,5]^2, T = 1000, P = 100, E = 10^5, \varepsilon = 10^{-5}$. Розовый цвет соответствует области хаоса, точнее неопределенности судьбы при заданных параметрах точности и глубины итераций. На (б) изображена гистограмма числа повторения уникальных периодов, меньших 40. В правом верхнем углу также расположена таблица «Период – число повторений» для первых 20 периодов

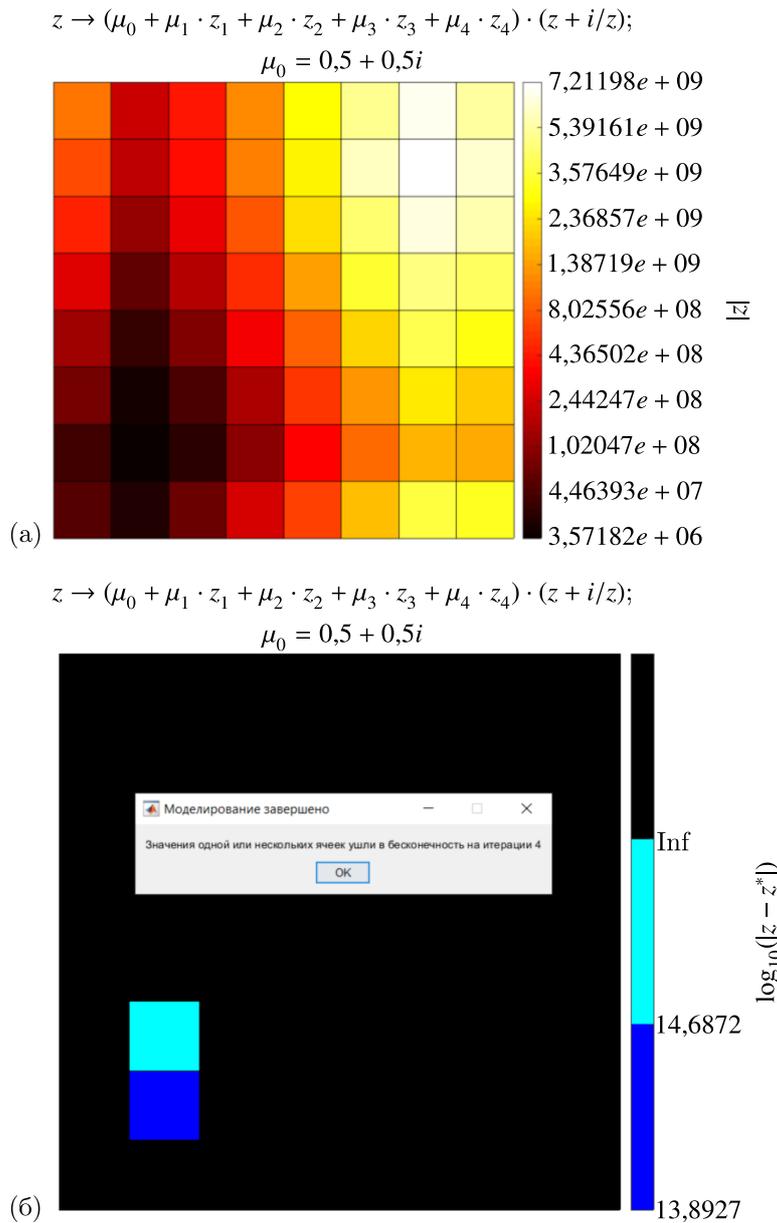


Рис. 12. Поля ячеек автомата на итерациях $T_1 = 3$ (а) и $T_2 = 5$ (б). Начальная конфигурация ячеек поля задана равномерным центральным распределением с параметрами $z_0 = 1, a = c = 1, b = 2. N = 8, E = 10^5, \varepsilon = 10^{-5}$

эксперименты в программе CMR даже в первой версии возможны на персональном компьютере с поддержкой многопоточности. Благодаря этому можно за разумное время оценить влияние связности (число соседей у внутренней ячейки: 4, 8 или 3, 6) и формы поля, а также типа границ на динамику клеточного автомата с локальной функцией перехода в виде заданного пользователем отображения.

Программа будет развиваться в направлении уточнения точек множеств Жюлиа (планируется устранить паллиатив с точками убегания и отдельно рассчитывать показатели Ляпунова). Кроме того, предполагается расширение инструментов анализа результатов КА-расчетов. Авторы надеются, что представленная программа будет использоваться математиками-профессионалами с обязательной ссылкой на данную статью.

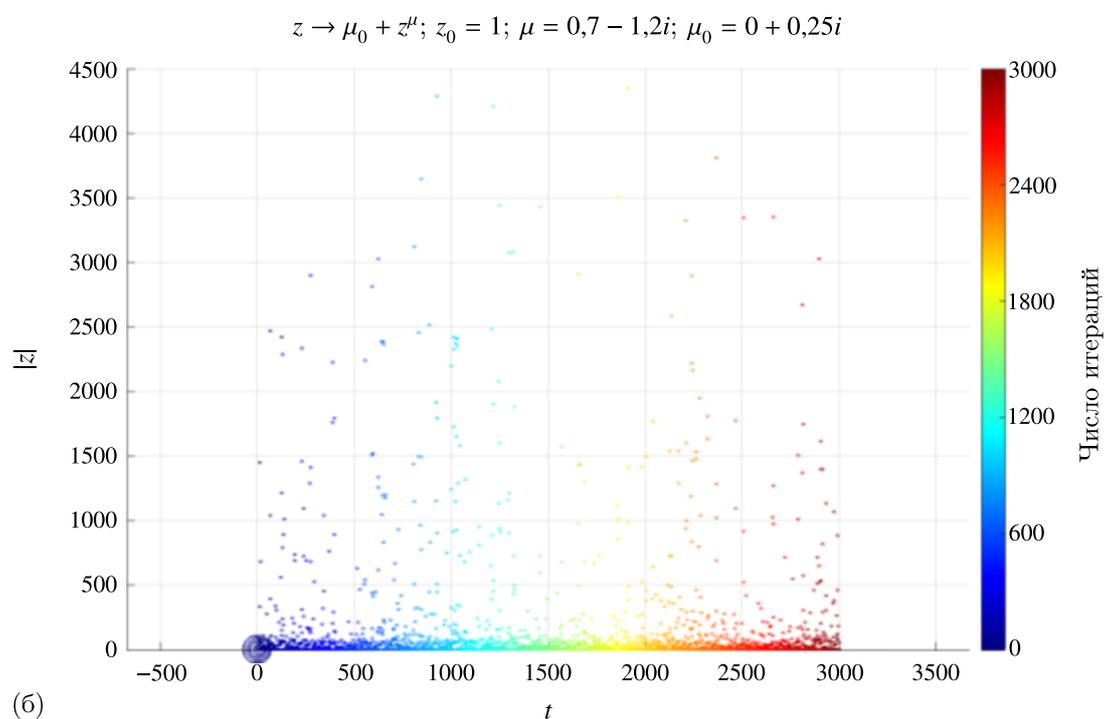
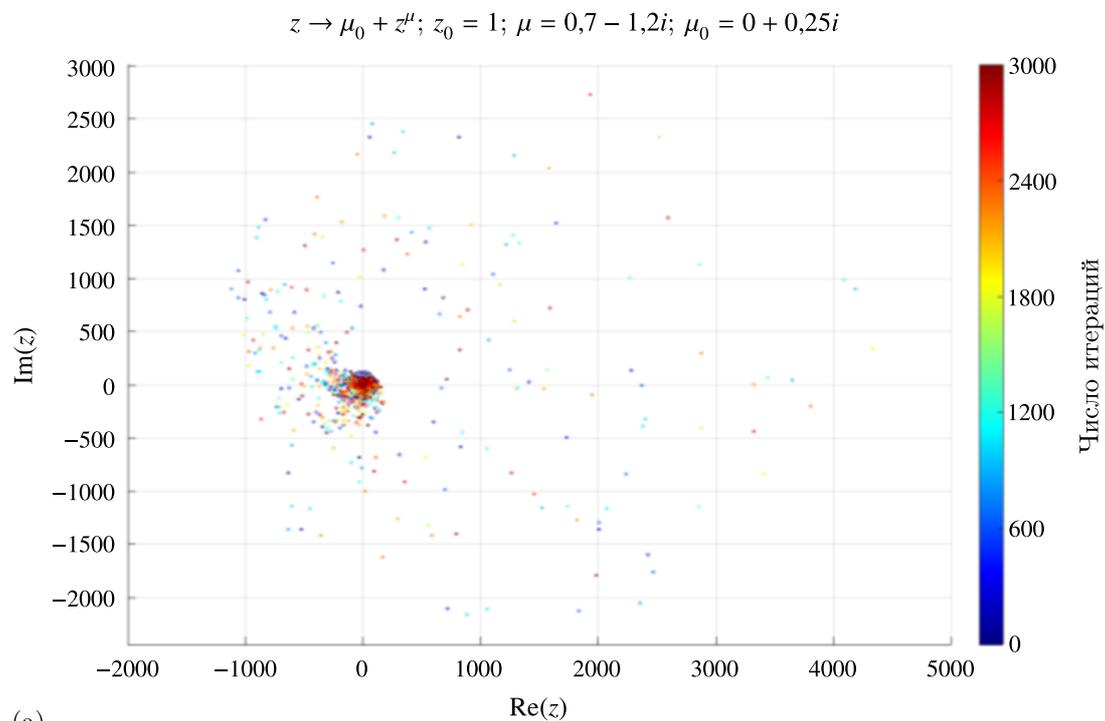


Рис. 13. Реальная и мнимая части (а) траектории точки $z_0 = 1$ в отображении $g: z \rightarrow \mu_0 + z^\mu$. На (б) визуализирована зависимость $|z|(t)$. Параметры отображения: $\mu = 0,7 - 1,2i, \mu_0 = 0,25i$. Настройки моделирования: $T = 3000, P = 1000, E = 10^5, \varepsilon = 10^{-5}$

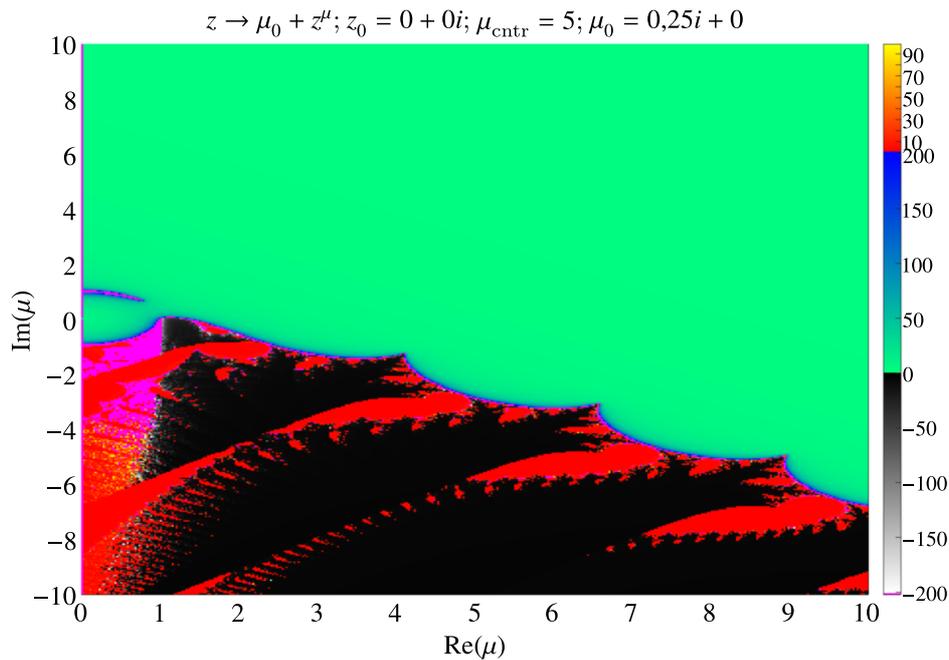


Рис. 14. Множество Мандельброта из «судеб» семейства отображений $g: z \rightarrow \mu_0 + z^{\mu}$ с параметрами $z_0 = 0, \mu_0 = 0,25i, \text{Im}(\mu) \in [-10; 10], \text{Re}(\mu) \in [0; 10]$. Настройки моделирования: $T = 200, P = 100, E = 10^5, \varepsilon = 10^{-5}$

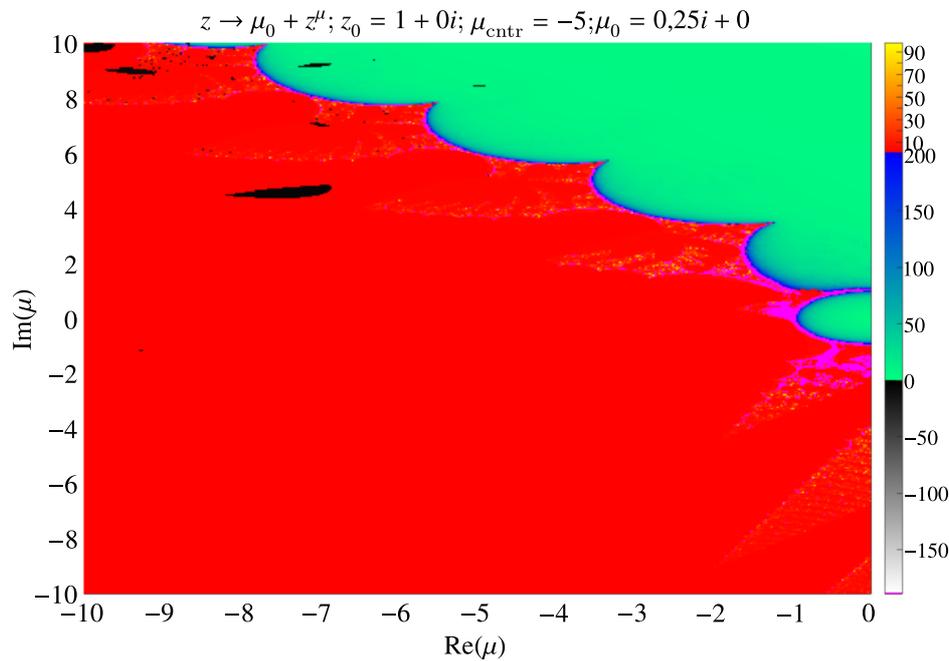


Рис. 15. Множество Мандельброта из «судеб» семейства отображений $g: z \rightarrow \mu_0 + z^{\mu}$ с параметрами $z_0 = 1 + 0i, \mu_0 = 0,25i, \text{Im}(\mu) \in [-10; 10], \text{Re}(\mu) \in [-10; 0]$. Настройки моделирования: $T = 200, P = 100, E = 10^5, \varepsilon = 10^{-5}$

Список литературы (References)

Арнольд В. И. Экспериментальная математика. — М.: ФАЗИС, 2005.
 Arnold V.I. Experimental mathematics / MSRI Mathematical Circles Library // Berkely, California, USA. American Mathematical Society, 2015. (Russ. ed.: Arnold V.I. Eksperimental'naja matematika. — М.: FAZIS, 2005.)

- Кулаков М. П., Фрисман Е. Я.* Бассейны притяжения кластеров в системах связанных отображений // *Нелинейная динамика*. — 2015. — Т. 11, № 1. — С. 51–76.
Kulakov M. P., Frisman E. Ya. Attraction basins of clusters in coupled map lattices // *Rus. J. Nonlin. Dyn.* — 2015. — Vol. 11, no. 1. — P. 51–76 (in Russian).
- Матюшкин И. В.* О некоторых свойствах отображения $\exp(iz)$ // *Нелинейная динамика*. — 2016. — Т. 12, № 1. — С. 3–15.
Matyushkin I. V. On some properties of an $\exp(iz)$ map // *Rus. J. Nonlin. Dyn.* — 2016. — Vol. 12, no. 1. — P. 3–15 (in Russian).
- Матюшкин И. В., Заплетина М. А.* Компьютерное исследование голоморфной динамики экспоненциального и линейно-экспоненциального отображений // *Компьютерные исследования и моделирование*. — 2018. — Т. 10, № 4. — С. 383–405.
Matyushkin I. V., Zapletina M. A. Computer research of the holomorphic dynamics of exponential and linear-exponential maps // *Computer research and modeling*. — 2018. — Vol. 10, no. 4. — P. 383–405 (in Russian).
- Поздняков М. В.* Динамические режимы и мультистабильность в системе несимметрично связанных двумерных отображений, демонстрирующих бифуркации удвоения Неймарка – Сакера // *Известия вузов. Прикладная нелинейная динамика*. — 2011. — Т. 19, № 4. — С. 68–76.
Pozdnyakov M. V. Dynamic regimes and multistability in the system of non-symmetrically coupled two-dimensional maps with period-doubling and Neimark – Sacker bifurcations // *Izvestiya VUZ. Applied Nonlinear Dynamics*. — 2011. — Vol. 19, iss. 4. — P. 68–76 (in Russian).
- Гитхуб-репозиторий программы Complex Mappings Researcher. — [Электронный ресурс]. — URL: https://github.com/PavelRubis/CA_Image (дата обращения: 13.09.2021).
- Alexander D., Devaney R. L.* A century of complex dynamics: A century of advancing mathematics / eds. S. F. Kennedy, D. J. Albers, G. L. Alexanderson, D. Dumbaugh, F. A. Farris, D. B. Haunspe. — Math. Assoc. America, Washington, D.C., 2015. — P. 15–34.
- «Experimental mathematics» journal. — URL: <https://www.tandfonline.com/toc/uexm20/current> (accessed: 13.09.2021).
- Feudel U.* Complex dynamics in multistable systems // *International Journal of Bifurcation and Chaos*. — 2008. — Vol. 18, № 6. — P. 1607.
- García-Morales V.* From deterministic cellular automata to coupled map lattices // *J. Phys. A: Math. Theor.* — 2016. — Vol. 49. — 295101. — 17 p.
- Hall G. M.* Adaptive code: Agile coding with design patterns and SOLID principles. — Microsoft press, 2017. — 762 p.
- Kaneko K.* Overview of coupled map lattices // *Chaos: An Interdisciplinary Journal of Nonlinear Science*. — 1992. — Т. 2, № 3. — P. 279–282.
- Matyushkin I. V., Zapletina M. A.* Experimental research of iterated dynamics for the complex exponentials with linear term // *Journal of Physics: Conference Series*. — 2018. — Vol. 990. — P. 012008. — DOI: 10.1088/1742-6596/990/1/012008
- Miritello G. et al.* Central limit behavior in the Kuramoto model at the «Edge of Chaos» // *Physica A*. — 2009. — Vol. 388. — P. 4818–4826.