Математика и математическое моделирование. 2019.
№ 03. С. 1 – 24.

DOI: **10.24108/mathm.0319.0000181**

Математика и Математическое моделирование

Сетевое научное издание

http://mathmelpub.ru ISSN 2412-5911

УДК 519.7

Клеточно-автоматный алгоритм пермутации матриц

Матюшкин И.В.[1,*], **Кожевников В.С.**[2]

[1]Национальный исследовательский университет «Московский институт электронной техники», Зеленоград, Москва, Россия

[2]Московский физико-технический институт (национальный исследовательский университет), Долгопрудный, Россия

*imatyushkin@niime.ru

В формализации клеточных автоматов (КА) описывается алгоритм, перемешивающий элементы матрицы на основе циклических сдвигов строк и столбцов. Он показывает интересное поведение при нечетном порядке матрицы n, при котором после пермутаций матрица претерпевает поворот на $\pm 90°$ и на $180°$ (отражение относительно центра), а рост периода N оказывается быстрее экспоненциального. Доказано, что N равно наименьшему общему кратному всех нечетных чисел, не превосходящих $2n$, т.е. $N = \text{НОК}(3,5,\text{К},2n-1)$. Динамика пермутаций анализируется с помощью введенных авторами двух «метрик», отражающих степень перемешанности. Результаты работы могут быть использованы при генерации псевдослучайных чисел.

Ключевые слова: клеточные автоматы, пермутация, перестановка, псевдослучайные числа, криптография, метрика

Представлена в редакцию: 06.05.2019, исправлена 21.05.2019

Введение

Пермутация, или перестановка элементов, матрицы — это один из способов генерации псевдослучайного числа. Простейшими пермутациями являются поворот, отражение и транспонирование матрицы [1, 2, 3]. Если элементы матрицы булевы или являются последовательностью нулей и единиц, то вся матрица отождествляется с двоичной записью некоторого числа, причем интерпретация зависит от способа преобразования двумерного индекса в одномерный.

Актуальность задачи поиска эффективных способов псевдослучайной пермутации матрицы обусловлена тем, что эта операция находит множество применений в различных областях. Генерация тестовых последовательностей в логических блоках современных

сверхбольших интегральных схем (СБИС) может требовать случайной перестановки данных внутри матричного блока, например, схемы с программируемой логикой (ПЛИС).

Помимо генерации случайных чисел перестановка матричных элементов непосредственно используется в криптографии для шифрования изображений [4, 5, 6, 7]. Информация, содержащаяся в цифровых изображениях, обычно отличается высокой избыточностью, а также наличием сильной корреляции между соседними пикселями. Данное обстоятельство предъявляет особые требования к используемому методу шифрования, удовлетворить которым позволяет комбинирование диффузии и конфузии, реализованное в виде перемешивающего преобразования [8]. При составлении стойких криптографических алгоритмов по сей день руководствуются утверждением Шеннона о том, что хорошее перемешивающее преобразование часто является результатом многократного применения произведения двух простых некоммутирующих операций [7, 8]. В данной работе в качестве таких операций могут рассматриваться сдвиги строк и столбцов матрицы.

Применение клеточных автоматов (КА) для генерации случайных последовательностей [9] и в криптографии [10, 11] вообще известно достаточно давно. Исследуемый в данной работе алгоритм можно использовать в качестве источника детерминированного хаоса, если глобальную функцию перехода соответствующего КА рассматривать как дискретизированное нелинейное отображение, как, например, преобразование пекаря [7], отображение «кот Арнольда» [5], сдвиг Бернулли [5] и т.п.

Цель работы заключается в разработке и исследовании работы КА-алгоритма на основе применения двух «метрик перемешанности» к проходимой КА цепочке состояний. Выполненное исследование также представляет интерес для комбинаторики и метрической теории для пермутаций двумерных объектов.

1. КА-алгоритм

Квадратная матрица $n \times n$ помещена в поле КА с замыканием границ (тороидальная решётка). Предлагается алгоритм преобразования матрицы с использованием циклических сдвигов строк и столбцов. Рассмотрим его на качественном уровне (рис. 1). Алгоритм состоит из 4 периодически повторяющихся этапов. Преобразование матрицы за один этап будем называть *шагом* или *тактом*, а за все 4 этапа — *проходом*.

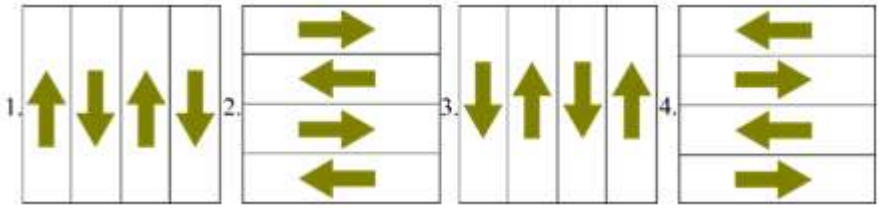

Рис. 1. Визуализация алгоритма

4 этапа алгоритма:

1. Нечётные столбцы сдвигаются вверх, чётные — вниз.
2. Нечётные строки сдвигаются вправо, чётные — влево.
3. Нечётные столбцы сдвигаются вниз, чётные — вверх.
4. Нечётные строки сдвигаются влево, чётные — вправо.

Приведём формальное описание алгоритма в терминах КА. Используется квадратная решётка $n \times n$ с замкнутыми границами (нумерация, традиционная для индексов матрицы). Окрестность клетки типа фон Неймана радиуса единица. Состояние клетки определяется двумя компонентами $\langle f, s \rangle$, см. табл. 1.

Таблица 1. Описание компонент клетки $\langle f, s \rangle$

Название	Флаг перемещения	Регистр данных
Обозначение	f	s
Множество значений	$\{\uparrow, \rightarrow, \downarrow, \leftarrow\} \equiv \{0,1,2,3\}$	Произвольное
Описание	Отвечает за перемещение элементов матрицы. Стрелки в основном указывают направление движения.	Хранит элементы матрицы M=$\|\|m_{ij}\|\|$

Начальное состояние автомата для матрицы (рис. 2):

$$\forall i, j \in \overline{1,n} : s_{ij} = m_{ij}$$

$$\begin{cases} j \bmod 2 = 1 \Rightarrow f_{ij} = \uparrow \\ j \bmod 2 = 0 \wedge i \bmod 2 = 1 \Rightarrow f_{ij} = \downarrow \\ j \bmod 2 = 0 \wedge i \bmod 2 = 0 \Rightarrow f_{ij} = \leftarrow \end{cases}$$

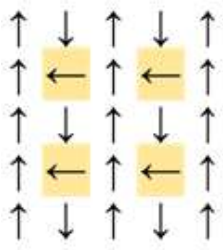

Рис. 2. Начальное состояние флага f при $n = 5$

Для формулировки локальной функции перехода (ЛФП) введем понятие согласованности. Две клетки называются *согласованными*, если чётности их флагов совпадают. Число согласованности *cons* клетки определяется как число согласованных с ней соседей, т.е. $cons = \left| \{ \mu \,|\, f_\mu \equiv f \pmod 2 \} \right|$, где μ — локальный индекс клетки. Его значение, как и f, можно указать стрелкой. Например, «$\uparrow$» — сосед сверху, «$\rightarrow$» — сосед справа, а $f_\downarrow$ — значение флага клетки снизу. Саму центральную клетку индексировать не будем. Знак «:=» имеет смысл присвоения значения, т.е. операнд справа от него берётся до перехода, а слева — после перехода. Операция mod возвращает остаток от целочисленного деления (например, $9 \bmod 4 = 1$, $-1 \bmod 4 = 3$)

Формальные правила перехода приведены ниже, в табл. 2. Нетрудно видеть, что по флагу f данный КА является циклическим. Блок правил 1 отвечает за перемещение элементов матрицы, а блок 2 — за изменение флага f.

Таблица 2. ЛФП алгоритма

№	Условие перехода	Формула перехода
1	$((cons=0)\wedge(f=0))\vee((cons\neq 0)\wedge(f=3))$	$s:=s_{\rightarrow}$
	$((cons=0)\wedge(f=1))\vee(cons\neq 0)\wedge(f=0)$	$s:=s_{\downarrow}$
	$((cons=0)\wedge(f=2))\vee((cons\neq 0)\wedge(f=1))$	$s:=s_{\leftarrow}$
	$((cons=0)\wedge(f=3))\vee((cons\neq 0)\wedge(f=2))$	$s:=s_{\uparrow}$
2	$(cons=0)\vee(cons=4)$	$f:=(f+1)\bmod 4$
	$\neg((cons=0)\vee(cons=4))$	$f:=(f-1)\bmod 4$

2. Результат работы алгоритма

Эмпирически установлено, что через некоторое число проходов алгоритм переводит матрицу в исходное состояние. Это позволяет назвать *периодом алгоритма* N минимальное (ненулевое) число проходов, через которое начальная конфигурация, содержащая попарно различные элементы, повторяется. Ясно, что такое определение не зависит от выбранной с этим условием начальной конфигурации.

Примечательным является период для матриц нечётных порядков. Для чётных порядков период линеен и равен $n/2$, а для нечётных имеет скорость роста, превышающую экспоненциальную. В табл. 3 представлены периоды для некоторых порядков n.

Таблица 3. Зависимость периода алгоритма $N(n)$ от порядка матрицы

Порядок матрицы n	Период N	Каноническое разложение N
3	15	3·5
5	315	3·3·5·7
7	45045	3·3·5·7·11·13
9	765765	3·3·5·7·11·13·17
11	14549535	3·3·5·7·11·13·17·19

Вычисление последней точки ($n=11$) занимает около двух часов на обычном компьютере. Следующая точка ($n=13$) потребовала бы около 200 часов, т.е. необходимы параллельные вычисления на суперкомпьютере. Несмотря на быстрый рост периода, число пройденных КА перестановок ничтожно мало по сравнению с их общим числом $n^2!$. Уже при $n=3$ просматривается лишь $1{,}6\cdot 10^{-4}$ от общего числа. Несмотря на короткую длину полученного ряда (табл. 3), обратим внимание, что нечётные простые множители p присутствуют все и входят в каноническое разложение N в первой степени. Первоначально мы предположили, что $N=\frac{3}{2}\prod_{p<2n} p,\quad n>3$, но затем уточнили эту гипотезу: N есть наименьшее общее кратное чисел $3,5,\text{K}\ ,2n-1$.

Рассмотрим теперь траектории отдельных элементов матрицы (рис. 3). Для этого поле с замкнутыми границами удобно представить не как тороидальную решётку, а как бесконечную плоскую периодическую решётку. Такое представление будем называть *развёр-*

нутым. Каждая из траекторий напоминает путь бильярдного шара, а граница поля играет роль стенки. Траектории коротко-периодичны, и их период зависит от начального положения элемента (в табл. 4 представлена эта зависимость для матрицы 13×13). Соответственно период работы алгоритма равен произведению индивидуальных периодов без учета разложения составных чисел на простые. Это в целом объясняет данные табл. 3. Обратим внимание, что возможны самопересечения траектории, когда позиция элемента совпадает с начальной, но направление его движения отлично от исходного.

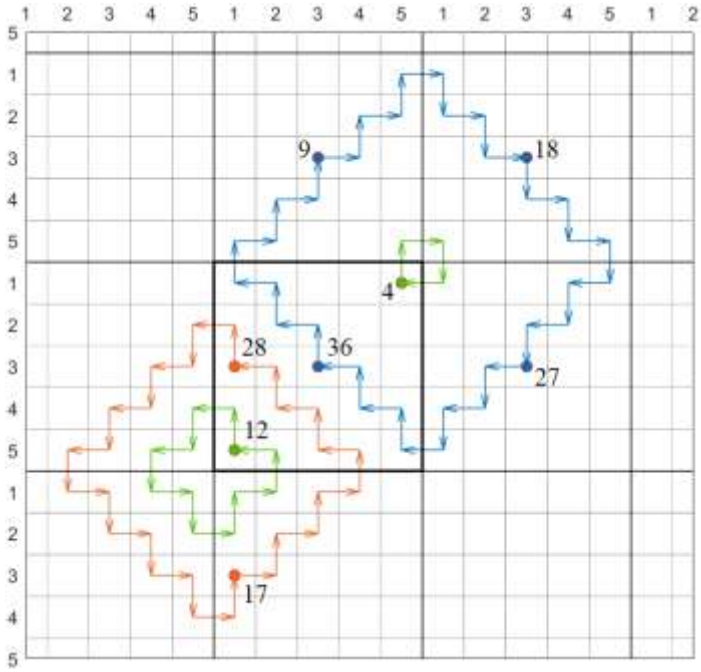

Рис. 3. Траектории некоторых элементов матрицы 5×5 в развёрнутом представлении. Исходная матрица — центральный квадрат. Точками отмечены начальные положения элементов, а для периферийных квадратов — места самопересечений траектории. Рядом с каждой точкой указано количество шагов до её достижения элементом

Таблица 4. Периоды индивидуальных траекторий. В клетки таблицы записаны номера шагов, на которых происходит возвращение соответствующих элементов матрицы 13×13 в исходную позицию

26/100	20	84	36	68	52	27/52	68	36	84	20	100	4
12	70/100	28	92	44	76	35/60	60	76	44	92	28	100
92	36	34/100	52	84	68	43/68	84	52	100	36	92	20
28	84	44	62/100	60	92	51/76	76	92	60	100	44	84
76	52	92	68	42/100	84	59/84	100	68	92	52	76	36
44	68	60	84	76	54/100	67/92	92	100	76	84	60	68
33/60	41/68	49/76	57/84	65/92	73/100	25/50/ 75/100	25/92	25/84	25/76	25/68	25/60	25/52
60	52	76	68	92	84	27/100	46/100	84	92	68	76	52
44	84	60	100	76	92	27/92	76	58/100	60	84	44	68
76	36	92	52	100	68	27/84	84	68	38/100	52	92	36
28	100	44	92	60	76	27/76	60	92	44	66/100	28	84
92	20	100	36	84	52	27/68	68	52	84	36	30/100	20
12	92	28	76	44	60	27/60	44	76	28	92	12	74/100

Последний номер равен индивидуальному периоду и дает следующие значения (в проходах), считая по строкам и без повторов: 25, 5, 21, 9, 17, 13, 1, 3, 7, 23, 11, 19, 15. Таким образом, ожидается $N(13) = 3^2 \cdot 5^2 \cdot 7 \cdot 11 \cdot 13 \cdot 17 \cdot 19 \cdot 23$.

Обоснование формулы $N = \text{НОК}(3,5,\text{K},2n-1)$ для произвольного нечётного $n \geq 3$ проводится по следующему плану:

1. До пересечения границы поля элементы двигаются параллельно главной и побочной диагоналям поля, а точнее, лесенкой по четырём схемам: вверх-влево $\left(\leftarrow\uparrow\right)$, вверх-вправо $\left(\uparrow\rightarrow\right)$, вниз-влево $\left(\leftarrow\downarrow\right)$, вниз-вправо $\left(\downarrow\rightarrow\right)$. Вертикальное направление зависит от четности номера начального столбца j, а горизонтальное — начальной строки i.

Таблица 5. Схема перемещения элемента в зависимости от начальных номеров строки i и столбца j

i \ j	нечётное	чётное
нечётное	$\leftarrow\uparrow$	$\leftarrow\downarrow$
чётное	$\uparrow\rightarrow$	$\downarrow\rightarrow$

2. При пересечении горизонтальной границы поля меняется направление движения по горизонтали (например, $\uparrow\rightarrow$ изменится на $\leftarrow\uparrow$), при пересечении вертикальной границы — по вертикали ($\uparrow\rightarrow$ изменится на $\downarrow\rightarrow$).
3. Период N_{ij} отдельного элемента с начальными индексами (i,j) явно выражается через положение, занимаемое данным элементом сразу после первого пересечения границы, и направление движения в этот момент.
4. Период N_{ij} нечётный и меньше $2n$.
5. Для каждого нечётного числа от 3 до $2n-1$ существует элемент с таким периодом.
6. $N(n) = \text{НОК}\left\{N_{ij}(n)\right\}_{i,j=1}^{n} = \text{НОК}(3,5,\text{K},2n-1)$.

Последовательно докажем каждый пункт.

1. Из определения алгоритма следует, что вертикальные и горизонтальные перемещения каждого элемента чередуются. Покажем, что вертикальное и горизонтальное направления перемещения сохраняются (до пересечения границы). Иначе говоря, если элемент переместился вверх, то через шаг он опять переместится вверх; если он переместился влево, то через шаг опять переместится влево, и т.д. В самом деле, так как направление движения столбца, в котором изначально находился элемент, через шаг изменится (рис. 1), то в соседних столбцах направление будет совпадать с исходным. Но в результате горизонтального перемещения на втором шаге рассматриваемый элемент и окажется в одном из соседних столбцов. Аналогично доказывается сохранение горизонтального направления. Это можно выразить формулами (1):

$$\begin{cases} i(t+2) = i(t) + \Delta i \\ j(t+2) = j(t) + \Delta j \end{cases} \tag{1}$$

где $i(t)$ — строка, в которой находится элемент на шаге t; $j(t)$ — столбец; $\Delta i, \Delta j = \pm 1$ — направления вертикального и горизонтального движения. Отсюда, с учётом очерёдности движения строк и столбцов, следуют выражения (2):

$$\begin{cases} i(t) = i_0 + \Delta i_0 \cdot \left[\dfrac{t+1}{2} \right] \\ j(t) = j_0 + \Delta j_0 \cdot \left[\dfrac{t}{2} \right] \end{cases} \tag{2}$$

где (i_0, j_0) — начальные индексы элемента; $\Delta i_0, \Delta j_0$ — направления движения до пересечения границы; $[x]$ — целая часть числа, $x - 1 < [x] \le x$.

Как следует из формул (2), существует шаг t_1, на котором i и j либо превышают n, либо становятся меньше 1, что соответствует пересечению границы поля (переходу в соседний квадрат в расширенном представлении). Так как после пересечения соотношения (1) остаются справедливыми, то при $t > t_1$ имеют место формулы, аналогичные (2). В процессе движения может происходить множество пересечений, поэтому обозначим шаг, на котором происходит k-е пересечение, через t_k. Тогда при $t_k \le t \le t_{k+1}$ движение описывается формулами (3):

$$\begin{cases} i(t) = i_k + \Delta i_k \cdot \left(\left[\dfrac{t+1}{2} \right] - \left[\dfrac{t_k+1}{2} \right] \right) \\ j(t) = j_k + \Delta j_k \cdot \left(\left[\dfrac{t}{2} \right] - \left[\dfrac{t_k}{2} \right] \right) \end{cases}, \quad t_k \le t \le t_{k+1} \tag{3}$$

где (i_k, j_k) — индексы элемента сразу после k-го пересечения границы, а $\Delta i_k, \Delta j_k$ — новые направления движения. Положим по определению $t_0 = 0$, тогда формулы (3) будут верны при всех $t \ge 0$, обобщая формулы (2).

Из определения алгоритма следует, что на первом шаге каждый элемент переместится вертикально, причём, в случае нечётного номера столбца j_0 — вверх ($\Delta i_0 = -1$), а в случае чётного — вниз ($\Delta i_0 = 1$). При этом, если номер строки i нечётный, то элемент окажется в чётной строке, а значит на втором шаге переместится влево ($\Delta j_0 = -1$). Соответственно, если номер i чётный, то элемент переместится вправо ($\Delta j_0 = 1$). Итак,

$$\begin{cases} \Delta i_0 = (-1)^{j_0} \\ \Delta j_0 = (-1)^{i_0} \end{cases} \tag{4}$$

2. Здесь принципиальна нечётность порядка матрицы n, из которой следует, что граничные столбцы и строки нечётные, а следовательно, имеют одинаковое направление движения. Пусть на шаге $t = t_k - 1$ элемент горизонтально двигался в строке $i(t_k - 1) \in \{1, n\}$, а на шаге $t = t_k$ элемент пересекает горизонтальную границу поля, т.е. переходит в строку $i_k = n + 1 - i(t_k - 1)$. Тогда направление горизонтального перемещения строки $i(t_k - 1)$ на шаге $t = t_k + 1$ будет противоположным направлению на шаге $t = t_k - 1$, но оно совпадает с направлением движения строки i_k, так как и $i(t_k - 1)$, и i_k нечётные. Аналогично рассматривается случай пересечения вертикальной границы поля. Итак, если была пересечена горизонтальная граница, то $\Delta i_k = \Delta i_{k-1}$, $\Delta j_k = -\Delta j_{k-1}$, если же вертикальная, то $\Delta i_k = -\Delta i_{k-1}$, $\Delta j_k = \Delta j_{k-1}$.

3. Мы показали, что до пересечения границы поля движение по горизонтали и по вертикали происходит в одном направлении, т.е. номера текущей строки и столбца монотонно изменяются, Сразу сделаем замечание, что формулы (2) имеют смысл лишь при $1 \le i(t), j(t) \le n$, а также в случае, когда один из индексов принимает значение 0 или $n+1$, что отвечает пересечению границы поля.

Рассмотрим движение элемента после пересечения горизонтальной границы на некотором шаге t_k. Сразу можно сказать, что t_k нечётно, так как горизонтальная граница может быть пересечена только при вертикальном перемещении, которое всегда происходит на нечётном шаге. Номер строки после пересечения может иметь два значения: $i_k \in \{1, n\}$. $i_k = 1$ при движении вниз, т.е. при $\Delta i_k = \Delta i_{k-1} = 1$, и $i_k = n$ при движении вверх, т.е. при $\Delta i_k = \Delta i_{k-1} = -1$. Это можно выразить формулой

$$i_k = \frac{1-n}{2}(1 + \Delta i_k) + n, \tag{5}$$

которая пригодится в дальнейшем. Покажем, что следующая граница поля, которую элемент пересечёт на шаге t_{k+1}, будет вертикальная. t_{k+1} чётно, так как горизонтальные перемещения происходят только на чётных шагах. Тогда при $t = t_{k+1}$ формулы (3) принимают вид

$$\begin{cases} i(t_{k+1}) = i_k + \Delta i_k \cdot \left(\dfrac{t_{k+1} - t_k - 1}{2} \right) \\ j(t_{k+1}) = j_k + \Delta j_k \cdot \left(\dfrac{t_{k+1} - t_k + 1}{2} \right) \end{cases} \tag{6}$$

Если на шаге t_{k+1} пересекается горизонтальная граница, то $i(t_{k+1}) = n + 1$ при $\Delta i_k = 1$ и $i(t_{k+1}) = 0$ при $\Delta i_k = -1$, откуда находим, что $t_{k+1} - t_k = 2n + 1$. Если же пересекается верти-

кальная граница, то $j(t_{k+1}) = n+1$ при $\Delta j_k = 1$ и $j(t_{k+1}) = 0$ при $\Delta j_k = -1$. Можно написать, что $j(t_{k+1}) = \frac{1+\Delta j_k}{2} \cdot (n+1)$, откуда

$$t_{k+1} - t_k = 2\frac{j(t_{k+1}) - j_k}{\Delta j_k} + 1 = n + \Delta j_k (n + 1 - 2j_k) = \begin{cases} 2n+1-2j_k, & \Delta j_k = 1 \\ 2j_k - 1, & \Delta j_k = -1 \end{cases} \quad (7)$$

Каким бы ни было $\Delta j_k = \pm 1$, имеет место неравенство $n + \Delta j_k (n+1-2j_k) < 2n+1$, т.е. вертикальное пересечение произойдёт раньше.

Подставляя выражения (7) и (5) в формулу (6), можно найти строку пересечения на шаге t_{k+1}:

$$i_{k+1} = i(t_{k+1}) = \frac{n+1+\Delta i_k \Delta j_k (n+1-2j_k)}{2} = \begin{cases} n+1-j_k, & \Delta i_k = \Delta j_k \\ j_k, & \Delta i_k \neq \Delta j_k \end{cases} \quad (8)$$

Номер столбца $j_{k+1} = 1$ при движении вправо, т.е. при $\Delta j_k = 1$, и $j_{k+1} = n$ при движении влево, т.е. при $\Delta j_k = -1$, что задаётся формулой (9):

$$j_{k+1} = \frac{1-n}{2}(1 + \Delta j_k) + n. \quad (9)$$

Как уже было установлено, при пересечении вертикальной границы меняет знак коэффициент Δi, следовательно

$$\begin{cases} \Delta i_{k+1} = -\Delta i_k \\ \Delta j_{k+1} = \Delta j_k \end{cases} \quad (10)$$

Итак, мы получили важные характеристики движения элемента после пересечения горизонтальной границы, такие как время (число шагов) до следующего пересечения границы, причём обязательно вертикальной, и индексы после пересечения.

Аналогично рассматривая пересечение вертикальной границы на некотором шаге t_k, можно установить, что следующая пересечённая граница будет горизонтальной, найти время до пересечения $t_{k+1} - t_k$ и столбец пересечения j_{k+1}. Приведём только результаты:

$$t_{k+1} - t_k = n + \Delta i_k (n + 1 - 2i_k) = \begin{cases} 2n+1-2i_k, & \Delta i_k = 1 \\ 2i_k - 1, & \Delta i_k = -1 \end{cases} \quad (11)$$

$$\begin{cases} i_{k+1} = \frac{1-n}{2}(1 + \Delta i_k) + n \\ j_{k+1} = \frac{n+1+\Delta i_k \Delta j_k (n+1-2i_k)}{2} = \begin{cases} n+1-i_k, & \Delta i_k = \Delta j_k \\ i_k, & \Delta i_k \neq \Delta j_k \end{cases} \end{cases} \quad (12)$$

$$\begin{cases} \Delta i_{k+1} = \Delta i_k \\ \Delta j_{k+1} = -\Delta j_k \end{cases} \quad (13)$$

Теперь, располагая формулами (7–13), мы имеем возможность проследить всю траекторию элемента с произвольными начальными индексами (i_0, j_0). Покажем, что после четырёх пересечений границы он вернётся в исходное положение. Для этого рассмотрим движение из клетки (i_1, j_1), т.е. после первого пересечения границы.

Если первая пересечённая граница горизонтальная, то из формул (7–13) следует:

а)

$$t_2 - t_1 = n + \Delta j_1 (n + 1 - 2 j_1)$$
$$i_2 = \frac{1}{2}\left(n + 1 + \Delta i_1 \Delta j_1 (n + 1 - 2 j_1)\right)$$
$$j_2 = \frac{1-n}{2}(1 + \Delta j_1) + n$$
$$\Delta j_2 = \Delta j_1$$
$$\Delta i_2 = -\Delta i_1$$

б)

$$t_3 - t_2 = n + \Delta i_2 (n + 1 - 2 i_2) = n - \Delta i_1 \left(n + 1 - (n+1) - \Delta i_1 \Delta j_1 (n + 1 - 2 j_1)\right) =$$
$$= n + \Delta j_1 (n + 1 - 2 j_1) = t_2 - t_1$$
$$i_3 = \frac{1-n}{2}(1 + \Delta i_2) + n = \frac{1-n}{2}(1 - \Delta i_1) + n$$
$$j_3 = \frac{1}{2}\left(n + 1 + \Delta i_2 \Delta j_2 (n + 1 - 2 i_2)\right) =$$
$$= \frac{1}{2}\left(n + 1 - \Delta i_1 \Delta j_1 \left(n + 1 - (n+1) - \Delta i_1 \Delta j_1 (n + 1 - 2 j_1)\right)\right) =$$
$$= \frac{1}{2}(n + 1 + n + 1 - 2 j_1) = n + 1 - j_1$$
$$\Delta i_3 = \Delta i_2 = -\Delta i_1$$
$$\Delta j_3 = -\Delta j_2 = -\Delta j_1$$

Так как $\Delta j_3 = -\Delta j_1$, то при $t \in [t_1, t_2)$ и при $t \in [t_3, t_4)$ горизонтальное движение происходит в разных направлениях, следовательно на шагах t_2 и t_4 пересекаются разные вертикальные границы (левая и правая).

в)

$$t_4 - t_3 = n + \Delta j_3 (n + 1 - 2 j_3) = n - \Delta j_1 \left(n + 1 - 2(n+1) + 2 j_1\right) = n + \Delta j_1 (n + 1 - 2 j_1) = t_2 - t_1$$
$$i_4 = \frac{1}{2}\left(n + 1 + \Delta i_3 \Delta j_3 (n + 1 - 2 j_3)\right) = \frac{1}{2}\left(n + 1 + \Delta i_1 \Delta j_1 \left(n + 1 - 2(n+1) + 2 j_1\right)\right) =$$
$$= (n+1)\frac{1 - \Delta i_1 \Delta j_1}{2} + \Delta i_1 \Delta j_1 \cdot j_1$$
$$j_4 = \frac{1-n}{2}(1 + \Delta j_3) + n = \frac{1-n}{2}(1 - \Delta j_1) + n$$
$$\Delta i_4 = -\Delta i_3 = \Delta i_1$$
$$\Delta j_4 = \Delta j_3 = -\Delta j_1$$

Так как $\Delta i_4 = \Delta i_1 = -\Delta i_2$, то при $t \in [t_2, t_3)$ и при $t \in [t_4, t_5)$ вертикальное движение происходит в разных направлениях, следовательно на шагах t_3 и t_5 пересекаются разные горизонтальные границы (верхняя и нижняя). Таким образом, при движении все четыре границы поля пересекаются поочерёдно.

г)

$$t_5 - t_4 = n + \Delta i_4 (n + 1 - 2i_4) = n + \Delta i_1 \left(n + 1 - (n+1)(1 - \Delta i_1 \Delta j_1) - 2\Delta i_1 \Delta j_1 \cdot j_1\right) =$$
$$= n + \Delta i_1 ((n+1)\Delta i_1 \Delta j_1 - 2 j_1 \cdot \Delta i_1 \Delta j_1) = n + \Delta j_1 (n + 1 - 2 j_1) = t_2 - t_1$$

$$i_5 = \frac{1-n}{2}(1 + \Delta i_4) + n = \frac{1-n}{2}(1 + \Delta i_1) + n$$

$$j_5 = \frac{1}{2}\left(n + 1 + \Delta i_4 \Delta j_4 (n + 1 - 2i_4)\right) =$$

$$= \frac{1}{2}\left(n + 1 - \Delta i_1 \Delta j_1 \left(n + 1 - (n+1)(1 - \Delta i_1 \Delta j_1) - 2\Delta i_1 \Delta j_1 \cdot j_1\right)\right) =$$

$$= \frac{1}{2}\left(n + 1 - \Delta i_1 \Delta j_1 \left(\Delta i_1 \Delta j_1 (n + 1 - 2 j_1)\right)\right) = \frac{1}{2}\left(n + 1 - (n+1) + 2 j_1\right) = j_1$$

По формуле (5) $i_5 = \frac{1-n}{2}(1 + \Delta i_1) + n = i_1$.

Если первой пересекается вертикальная граница, то рассуждения полностью аналогичны. В общем случае, для всех $k \in \overline{1,4}$ верно равенство:

$$t_{k+1} - t_k = n + \sigma(n + 1 - 2l),$$

где $\sigma = \Delta j_1$, $l = j_1$, если на шаге t_1 была пересечена горизонтальная граница, и $\sigma = \Delta i_1$, $l = i_1$, если вертикальная. При этом $i_5 = i_1$ и $j_5 = j_1$, т.е. элемент возвращается в положение (i_1, j_1) на шаге $t = t_1 + \tau$, где $\tau = 4(n + \sigma(n + 1 - 2l))$.

Покажем, что τ и есть период данного элемента. Для этого заметим, что алгоритм является обратимым, т.е. предыдущее состояние однозначно определяется текущим состоянием и номером шага по модулю 4, точнее числом $(t \bmod 4)$, что следует из определения алгоритма (4 этапа и цикличность). Так как $\tau \bmod 4 = 0$, то элемент возвращается в клетку (i_1, j_1) на том же этапе алгоритма, что и входит в неё на шаге t_1. Из обратимости алгоритма следует, что на шаге $t = (t_1 + \tau) - t_1 = \tau$ элемент находился в той же клетке, из которой он дошёл до (i_1, j_1) за t_1 шагов, т.е. в (i_0, j_0). Итак, элемент действительно возвращается в (i_0, j_0) за τ шагов, т.е. за $\tau / 4$ проходов.

Докажем, что на промежутке времени (измеряемого в шагах) $[1, \tau)$ нет шага, кратного 4, на котором элемент также проходил бы клетку (i_0, j_0). В самом деле, если бы на некотором шаге τ_x, таком что $\tau_x \bmod 4 = 0$, элемент оказался бы в (i_0, j_0), то первой границей, которую бы он пересёк, была бы та же, что и на шаге t_1, в силу цикличности алгоритма.

Но, так как пересекаемые границы чередуются, то пересечение этой границы возможно лишь на шагах t_1 и t_5. На шаге t_1 оно произойти не может, иначе $\tau_x \in [1, t_1)$, а, как следует из формул (2), при этом i и j изменяются монотонно от i_0 и j_0, следовательно, не могут вернуться к исходным значения до шага t_1. Таким образом, $\tau_x \in [t_4, t_5)$, но тогда $\tau_x = \tau$, так как на $[t_4, t_5)$ каждая клетка проходится элементом лишь один раз, потому что i и j опять же меняются монотонно, что следует из формул (3). Итак, τ — первый кратный 4 шаг, на котором происходит возвращение элемента в исходное положение. Для периода данного элемента можно написать формулу

$$N_{i_0 j_0} = n + \sigma(n + 1 - 2l), \tag{14}$$

где $\sigma = \Delta j_1$, $l = j_1$, если на шаге t_1 была пересечена горизонтальная граница, и $\sigma = \Delta i_1$, $l = i_1$, если вертикальная.

4. Нечётность N_{ij} следует из формулы (14), как и тот факт, что $N_{ij} \le 2n - 1$.

5. Найдём явную зависимость $N_{i_0 j_0}$ от (i_0, j_0), определив, как соотносятся (i_1, j_1) и (i_0, j_0). Для этого перепишем формулы (2) для шага t_1 в виде

$$\begin{cases} (i(t_1) - i_0) \cdot \Delta i_0 = \left[\dfrac{t_1}{2}\right] + t_1 \bmod 2 \\ (j(t_1) - j_0) \cdot \Delta j_0 = \left[\dfrac{t_1}{2}\right] \end{cases} \tag{15}$$

Вычитая одно равенство в (15) из другого, получаем

$$(i(t_1) - i_0) \cdot \Delta i_0 = (j(t_1) - j_0) \cdot \Delta j_0 + t_1 \bmod 2. \tag{16}$$

Далее следует рассмотреть различные случаи чётных и нечётных начальных индексов (i_0, j_0). Например, если $i_0 \bmod 2 = 0$ и $j_0 \bmod 2 = 0$, то, по формулам (4), $\Delta i_0 = 1$ и $\Delta j_0 = 1$.

Если $j(t_1) = n + 1$ (случай невозможен, так как $\Delta j_0 = 1 > 0$ и $j(t)$ возрастает), т.е. первой пересекается правая граница, то $t_1 \bmod 2 = 0$, то из уравнения (16) следует, что

$$i_1 = n + 1 + i_0 - j_0. \tag{17}$$

Так как $1 \le i_1 \le n$, то из полученной формулы (17) находим условие реализации данного случая:

$$j_0 - i_0 \ge 1.$$

Если же $i(t_1) = n + 1$, т.е. первой пересекается нижняя граница, то $t_1 \bmod 2 = 1$ и

$$j_1 = n + j_0 - i_0,$$

а так как $1 \le j_1 \le n$, то

$$j_0 - i_0 \le 0.$$

Остальные случаи рассматриваются аналогичным образом. Результаты сведены в табл. 6.

Таблица 6. Зависимость (i_1, j_1) и $N_{i_0 j_0}$ от (i_0, j_0)

Условие		i_1	j_1	$N_{i_0 j_0}$
$\begin{cases} i_0 \bmod 2 = 0 \\ j_0 \bmod 2 = 0 \end{cases}$	$j_0 \geq i_0 + 1$	$n+1+i_0-j_0$	1	$2(i_0-j_0)+2n+1$
	$j_0 \leq i_0$	1	$n+j_0-i_0$	$2(j_0-i_0+n)-1$
$\begin{cases} i_0 \bmod 2 = 1 \\ j_0 \bmod 2 = 1 \end{cases}$	$i_0 \geq j_0 + 1$	i_0-j_0	n	$2(j_0-i_0)+2n+1$
	$i_0 \leq j_0$	n	j_0-i_0+1	$2(i_0-j_0)+2n-1$
$\begin{cases} i_0 \bmod 2 = 0 \\ j_0 \bmod 2 = 1 \end{cases}$	$i_0+j_0 \geq n+2$	i_0+j_0-n-1	1	$4n+3-2(i_0+j_0)$
	$i_0+j_0 \leq n+1$	n	i_0+j_0-1	$2(i_0+j_0)-3$
$\begin{cases} i_0 \bmod 2 = 1 \\ j_0 \bmod 2 = 0 \end{cases}$	$i_0+j_0 \leq n$	i_0+j_0	n	$2(i_0+j_0)-1$
	$i_0+j_0 \geq n+1$	1	i_0+j_0-n	$4n+1-2(i_0+j_0)$

Как следует из предыдущего пункта, любой индивидуальный период N_{ij} нечётный , а значит представим в виде $N_{ij} = 2p-1$, $p \in \Gamma$. Нас интересует лишь случай $3 \leq N_{ij} \leq 2n-1$, или $p \in \overline{2,n}$. Подберём такие начальные индексы (i_0, j_0), что $N_{i_0 j_0} = 2p-1$ при заданном $p \in \overline{2,n}$. Несложно проверить, что при нечётном p подойдут значения

$$\begin{cases} i_0 = 1 \\ j_0 = n+1-p \end{cases}$$

При этом реализуется случай $N_{i_0 j_0} = 2(i_0-j_0)+2n-1 = 2p-1$. При чётном же p можно взять значения

$$\begin{cases} i_0 = n \\ j_0 = n+1-p \end{cases}$$

Тогда $N_{i_0 j_0} = 4n+1-2(i_0+j_0) = 2p-1$.

6. Период алгоритма кратен периоду каждого элемента. Так как по определению период алгоритма есть минимальное число проходов, за которое матрица возвращается в исходное состояние, то $N(n) = \text{НОК}\left\{N_{ij}(n)\right\}_{i,j=1}^{n}$. Среди $\left\{N_{ij}(n)\right\}_{i,j=1}^{n}$ есть лишь нечётные числа, не превышающие $2n-1$, причём все такие числа. Следовательно, $N(n) = \text{НОК}(3,5,\text{К},2n-1)$. Ч.т.д.

3. Используемые «метрики перемешанности»

Каждый ход КА дает нам пермутацию матрицы относительно начальной (здесь и далее считаем все элементы различными). Для описания динамики КА желательно иметь числовую характеристику, назовем её *mixedness*, «удалённости» текущей перестановки от начальной конфигурации. Существует большое число разнообразных метрик, определённых на множестве пермутаций массива. Среди них l_p-метрики, Хэммингова метрика, лексикографическая и другие, которые, однако, все зависят от значений элементов перестановок [12]. Здесь же предлагаются «метрики», характеризующие лишь взаимное расположение элементов. Мы будем использовать две «метрики»: матричную mm (matrix mixedness), которая наследована от манхэттенской метрики на плоскости и является метрикой в математическом смысле, и линейную lm (linear mixedness), которая адресуется массиву данных. Метрика mm формально схожа с l_1-метрикой, однако адаптирована специально для матричных перестановок. «Метрика» lm не является математической метрикой, но более эффективна по сравнению с mm в поиске особых конфигураций.

Пусть в некоторой матрице A порядка n записаны попарно различные элементы c_α, $\alpha \in \overline{1,n^2}$, а B является пермутацией A. Тогда определим метрику mm на этих двух матрицах формулой

$$mm = \frac{1}{n^2}\sum_{\alpha=1}^{n^2} s_\alpha ,$$

где $s_\alpha = |i_2 - i_1| + |j_2 - j_1|$, если (i_1, j_1) — это индексы элемента c_α в матрице A, а (i_2, j_2) — индексы c_α в матрице B. Выражение s_α имеет смысл расстояния между положениями элемента c_α в матрицах A и B по метрике Манхэттена.

Элементы c_α можно отождествить (поставить во взаимно-однозначное соответствие) с парами чисел (k,l), $k,l \in \overline{1,n}$, так, чтобы элементам матрицы $A = \|a_{kl}\|$ соответствовали пары их же индексов, например, через конструкцию $a_{kl} = k + n\cdot(l-1)$. После перестановки элемент a_{kl} окажется на позиции (i,j) в матрице $B = \|b_{ij}\|$, т.е. $b_{ij} = a_{kl} = (k,l)$. Например:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \equiv \begin{pmatrix} (1,1) & (1,2) & (1,3) \\ (2,1) & (2,2) & (2,3) \\ (3,1) & (3,2) & (3,3) \end{pmatrix} \to \begin{pmatrix} (1,1) & (2,2) & (3,1) \\ (2,1) & (3,2) & (2,3) \\ (1,3) & (1,2) & (3,3) \end{pmatrix} \equiv \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}$$

Тогда mm полученной матрицы вычисляется по формуле

$$mm = \frac{1}{n^2}\sum_{i,j=1}^{n} s_{ij} ,$$

где $s_{ij} = |k-i| + |l-j|$, если $b_{ij} = (k,l)$. Например, для приведённой выше матрицы третьего порядка $mm = \frac{1}{n^2}(s_{31} + s_{12} + s_{22} + s_{32} + s_{13}) = \frac{1}{9}(4+1+1+2+2) = \frac{10}{9}$.

Эмпирически выполнен поиск наибольшего значения mm при заданном n. В качестве начальной конфигурации были использованы либо исходная матрица, либо отраженная от центра. Алгоритм поиска простой:

1. Задаётся произвольная конфигурация матрицы.
2. Переставляются 2 элемента со случайными номерами.
3. Если при этом mm не возрастает, то перестановка элементов отменяется.
4. Повторяется пункт 2.

В диапазоне $1 \le n \le 40$ с числом итераций 10^4 не найдено конфигураций со значением mm, превышающим n.

«Метрика» lm вводится сначала для одномерного массива. Считая, что начальное его состояние имеет вид $a_k = k, \quad k \in \overline{1,m}$, lm-метрика определяется формулой

$$lm = \frac{m+1}{l_\uparrow + l_\downarrow},$$

где $l_\uparrow$ — максимальная длина возрастающей подпоследовательности, а $l_\downarrow$ — максимальная длина убывающей подпоследовательности. Числитель выбран так, чтобы lm исходного массива равнялась 1. В самом деле, для него $l_\uparrow = m$, а $l_\downarrow = 1$ (подпоследовательность из одного элемента).

Чтобы применить lm к матрице, необходимо сначала сопоставить матрице массив, например, записать её элементы в столбец. Это можно сделать различными способами, например, змейкой или спиралью. Мы будем использовать простую схему, заимствованную из MATLAB. Столбец получается последовательной конкатенацией всех столбцов матрицы, взятых в порядке возрастания индекса. Итак, по определению lm матрицы есть lm соответствующего ей столбца.

Можно ожидать, что пермутации с экстремальными значениями «метрики» особенные.

4. Анализ динамики КА алгоритма с помощью «метрик»

Ниже приведены графики mm и lm, соответствующие применению алгоритма к матрицам различных порядков. Точки, отвечающие особенным пермутациям, отмечены крестиком «×» и обозначены греческими буквами. Рассмотрим вначале четные порядки $n \bmod 2 = 0$. В качестве иллюстрации общего случая рассмотрим $n = 20$. Динамика обеих «метрик» показана на рис. 4 (с детализацией до такта (шага) t, а не прохода).

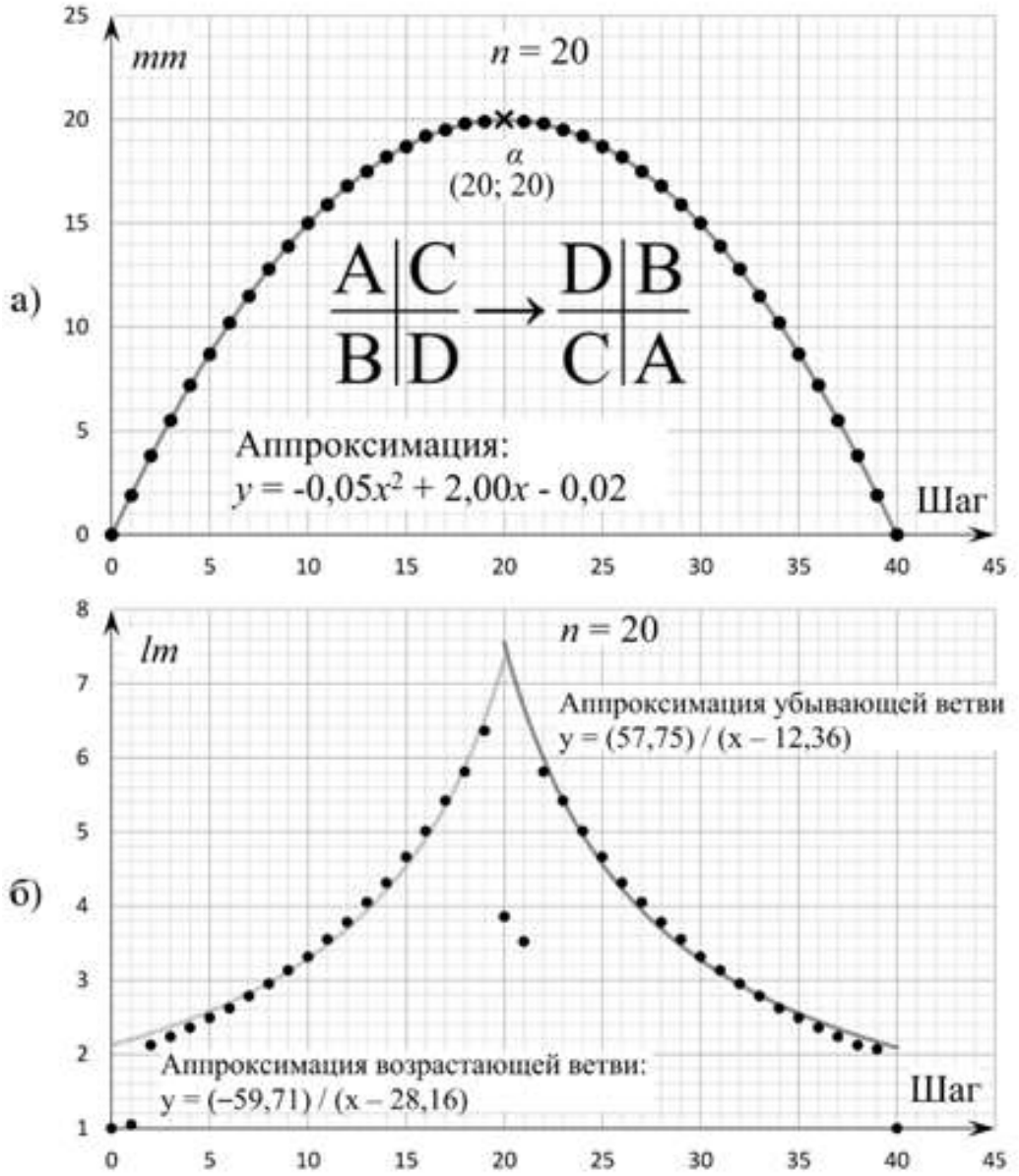

Рис. 4. Перемешанности последовательных конфигураций матрицы порядка $n = 20$, полученных применением алгоритма. В отличие от метрики mm (а) наблюдается асимметрия метрики lm (б), по-видимому, связанная с упорядоченностью элементов матрицы по столбцам, а не строкам, и выбором флагов первого такта

«Метрика» mm достигает максимума α на середине пути ($t = 20$), а lm в той же точке близка к локальному минимуму. Соответствующая конфигурация получена диагональной перестановкой блоков (рис. 4а); в силу четности n исходная матрица легко представляется в блочном виде. То же наблюдается и при других чётных порядках n.

Уравнение аппроксимирующей mm параболы имеет вид, близкий к $y = -\frac{x^2}{20} + 2x \to -\frac{x^2}{n} + 2x$. Оказывается, что на чётном шаге t метрика mm совпадает со значением, полученным по этой формуле. На нечётном же шаге реальное значение mm

меньше точно на $0.05 = \frac{1}{20} \to \frac{1}{n}$. Таким образом, для чётных n можно обобщить (равенство проверено для $n = 20$ и $n = 8$):

$$mm(t) = \begin{cases} -\frac{t^2}{n} + 2t, & t \bmod 2 = 0 \\ -\frac{t^2+1}{n} + 2t, & t \bmod 2 = 1 \end{cases}$$

«Метрика» lm в целом показывает нам две симметричные ветви, но эта симметрия нарушается в центре. Возможно, это связано с изначальной асимметрией движения столбцов в начальной конфигурации КА относительно порядка нумерации. Обе ветви аппроксимируются гиперболой $y = \frac{p}{x+q}$. Параметры аппроксимации MATLAB (значение, интервал значений, доверительная вероятность): для возрастающей ветви - $p = -59,71\ (-62,46; -56,97)$, $q = -28,16\ (-28,74; -27,58)$, $R^2 = 0,9942$; для убывающей ветви $p = 57,75\ (55,16; 60,33)$, $q = -12,36\ (-12,95; -11,76)$, $R^2 = 0,9945$.

Рассмотрим теперь случай нечетных порядков на двух примерах: $n = 5$ (рис. 5, 6) и $n = 7$ (рис. 7, 8). Дополнительно строим гистограммы «число пермутаций — значение метрики».

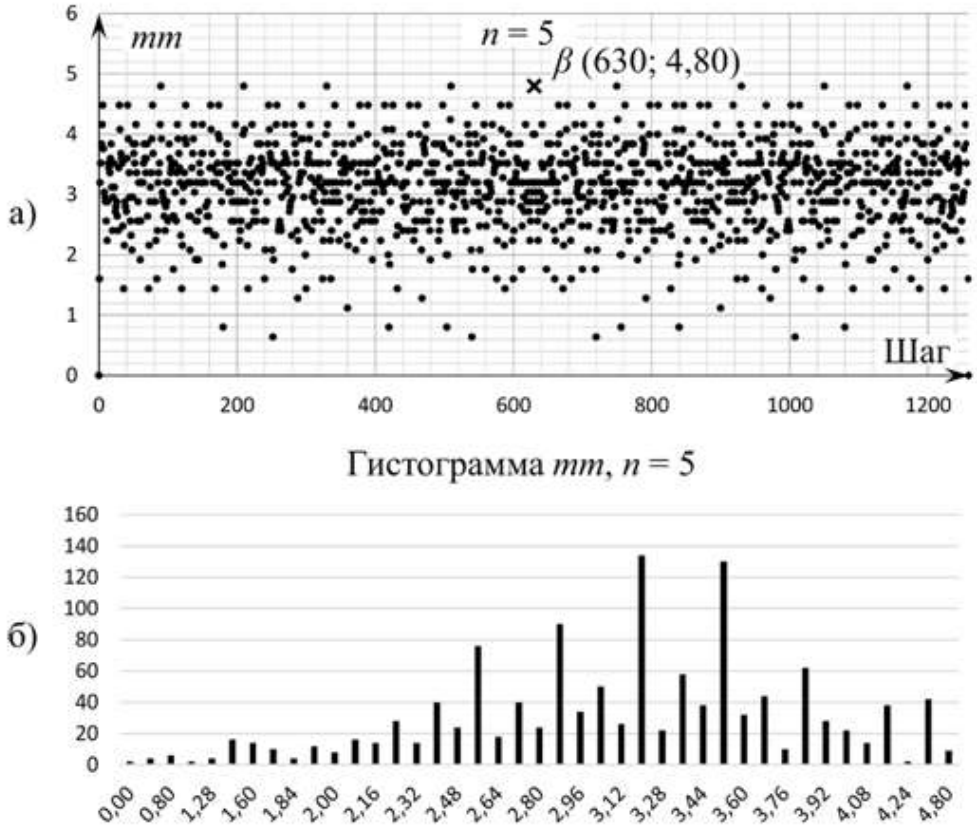

Рис. 5. Метрика mm последовательных конфигураций матрицы порядка $n = 5$, полученных применением алгоритма

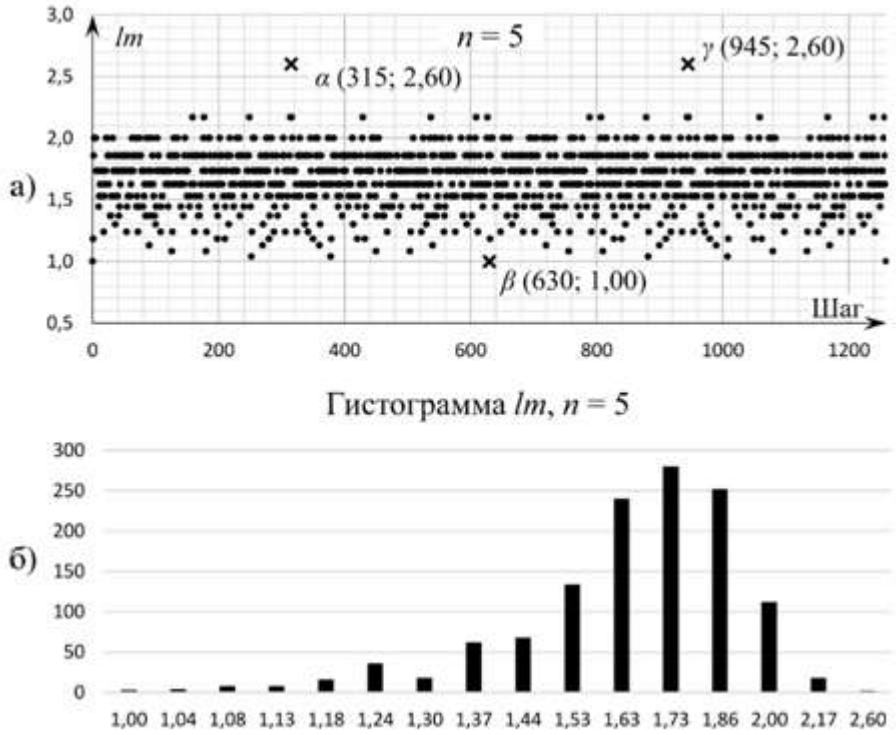

Рис. 6. «Метрика» lm последовательных конфигураций матрицы порядка $n=5$, полученных применением алгоритма

Матрица β соответствует одновременно глобальному максимуму mm и глобальному минимуму lm. Она представляет собой отражение исходной матрицы $\|a_{ij}\|$ относительно центра: $\beta_{ij}=a_{n+1-i,n+1-j}$ (элементы матрицы обозначены той же буквой, что и матрица, с индексами). Матрицы α и γ, на которых lm достигает глобального максимума, получены вращением $\|a_{ij}\|$ против и по часовой стрелке соответственно: $\alpha_{ij}=a_{j,n+1-i}$, $\gamma_{ij}=a_{n+1-j,i}$. Однако эти экстремумы mm-метрика «не замечает». Аналогичная ситуация наблюдается и при $n=7$.

Как и при $n=5$, матрица β есть отражение исходной относительно центра, а α и γ — повороты. Однако в данном случае α — поворот по часовой стрелке, а γ — против часовой стрелки. То есть при $n=5$ и $n=7$ алгоритм проходит эти конфигурации в разном порядке. Если же заметить, что отражение β есть ни что иное как поворот на $180°$, то получится, что алгоритм поворачивает матрицу на $90°$ за число шагов t_α, равное периоду N. Напомним, что алгоритм четырехтактовый, следовательно, он возвращает матрицу в исходное состояние за $4N$ шагов. Для $n=5$ вращение происходит против часовой стрелки, а для $n=7$ — по часовой стрелке. Примерное объяснение этому факту состоит в аналогии с волной.

Обращает на себя внимание асимметрия гистограмм и наличие, во всяком случае, визуальное, для $n=7$ и mm-метрики суперпозиции трех дискретных гауссовых распределений. Для бóльших порядков матрицы этот феномен требует более глубокого исследования.

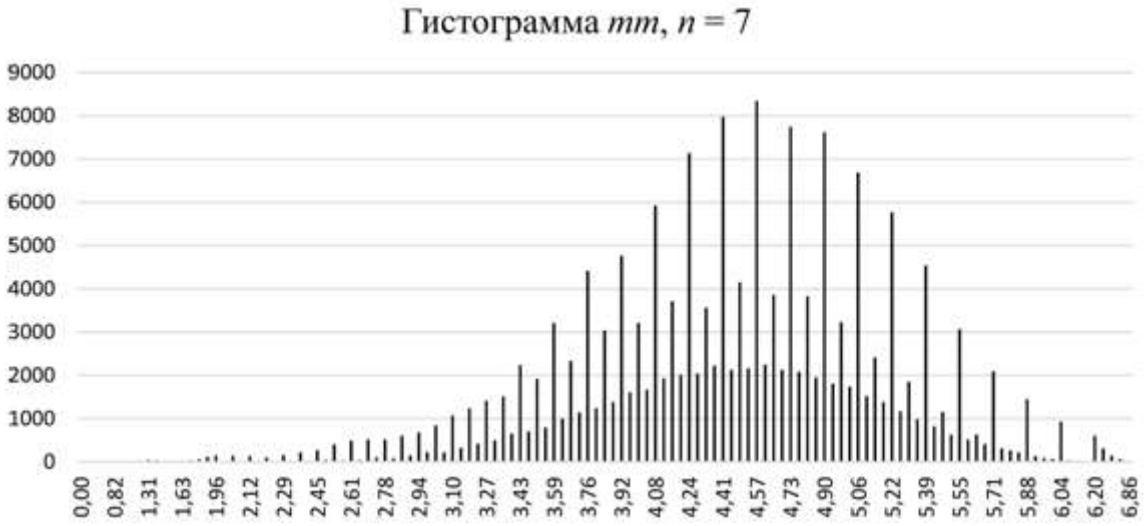

Рис. 7. Гистограмма метрики mm для матрицы порядка $n = 7$, полученных применением алгоритма

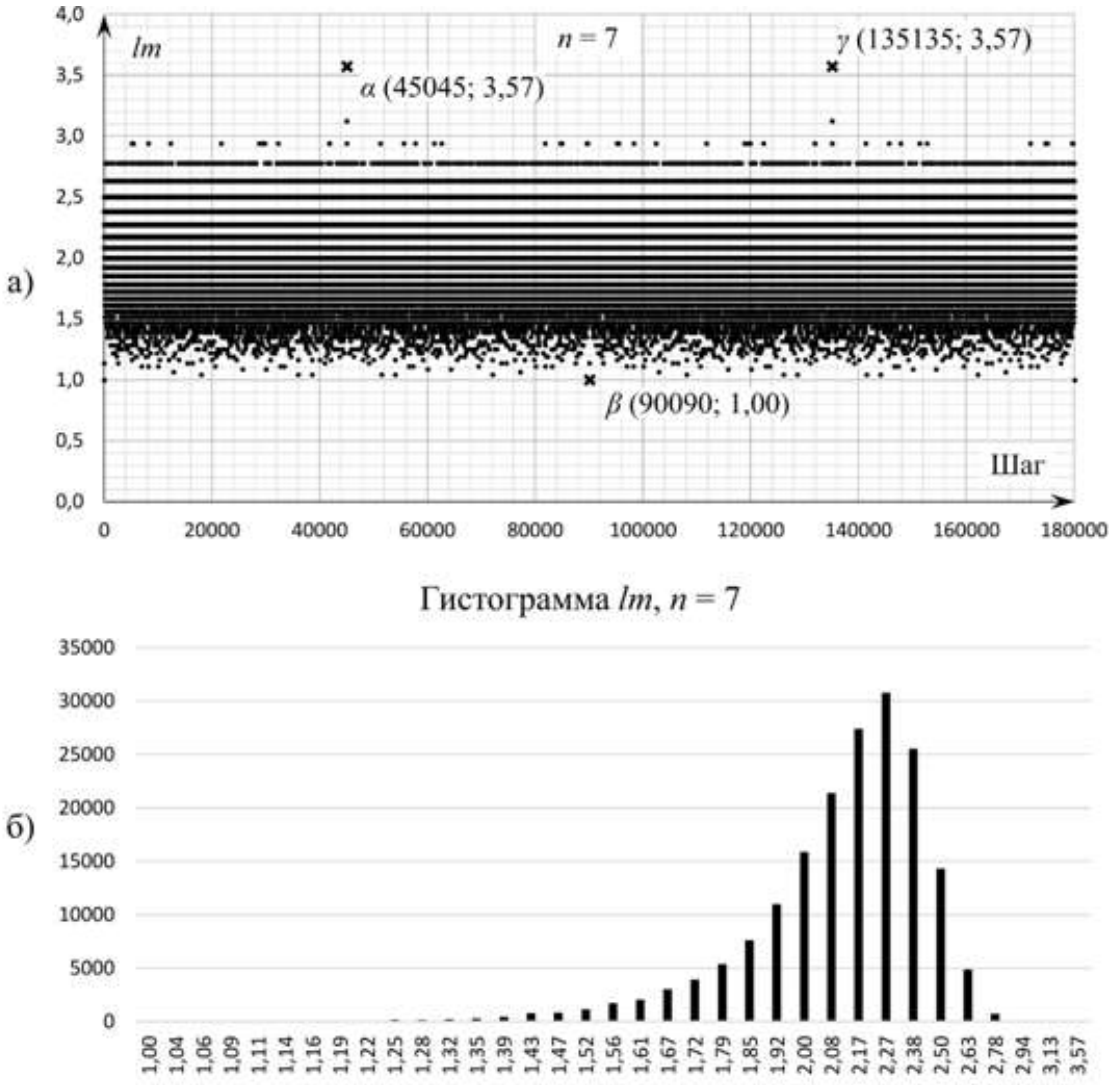

Рис. 8. «Метрика» (а) lm и её гистограмма (б) последовательных конфигураций матрицы порядка $n = 7$, полученных применением алгоритма

Заключение

В результате анализа алгоритма был выявлен ряд примечательных его свойств. В частности, последовательность его периодов N для нечётных порядков матриц n задаётся формулой $N(n) = \text{НОК}(3,5,\text{K},2n-1)$. Величина периода косвенно говорит в пользу алгоритма как генератора случайных чисел. Более того, период может быть существенно увеличен небольшим усложнением алгоритма, например, введением различных длин для циклов смены направления столбцов и строк. Останавливая алгоритм в случайный момент времени, текущую перестановку матрицы можно рассматривать как массив случайных чисел или же преобразовать её каким-либо способом в одно случайное число.

Установлено, что в ходе работы алгоритма отдельные элементы (рисунок 3) перемещаются аналогично бильярдным шарам под углом $45°$. Для усложнения динамики можно реализовать движение и под другим углом, например, $\text{arctg}(2)$.

Введённые «метрики перемешанности» позволили определить, что в результате применения алгоритма матрица претерпевает вращение в случае нечётного порядка и блочную перестановку в случае чётного. В результате численного эксперимента получена примечательная гистограмма для $n = 7$ (рис. 7), форма которой имеет вид трёх нормальных распределений, наложенных друг на друга. Изменение «метрик» mm и lm на последовательных конфигурациях матрицы имеет псевдослучайный вид (рис. 5, 6).

Список литературы

1. Матюшкин И.В. Алгоритмы параллельных вычислений в формализации клеточных автоматов: сортировка строк и умножение чисел по схеме Атрубина // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2016. № 4. С. 77–81.
2. Матюшкин И.В., Жемерикин А.В., Заплетина М.А. Клеточно-автоматные алгоритмы сортировки строк и умножения целых чисел по схеме Атрубина // Изв. высш. учеб. заведений. Электроника. 2016. Т. 21. № 6. С. 557–565.
3. Матюшкин И.В., Заплетина М.А. Отражение и транспонирование данных в матрице клеточно-автоматного вычислителя // Изв. высш. учеб. заведений. Электроника. 2019. Т. 24. № 1. С. 51–63. DOI: 10.24151/1561-5405-2019-24-1-51-63
4. Ji Won Yoon, Hyoungshick Kim. An image encryption scheme with a pseudorandom permutation based on chaotic maps // Communications in Nonlinear Science and Numerical Simulation. 2010. Vol. 15. No. 12. Pp. 3998–4006. DOI: 10.1016/j.cnsns.2010.01.041
5. Weichuang Guo, Junqin Zhao, Ruisong Ye. A chaos-based pseudorandom permutation and bilateral diffusion scheme for image encryption // Intern. J. of Image, Graphics and Signal Processing. 2014. Vol. 6. No. 11. Pp. 50–61. DOI: 10.5815/ijigsp.2014.11.07
6. Kaur M., Kumar V. Efficient image encryption method based on improved Lorenz chaotic system // Electronics Letters. 2018. Vol. 54. No. 9. Pp. 562–564. DOI: 10.1049/el.2017.4426

7. Fridrich J. Symmetric ciphers based on two-dimensional chaotic maps // Intern. J. of Bifurcation and Chaos in Applied Sciences and Engineering. 1998. Vol. 8. No. 6. Pp. 1259–1284. DOI: 10.1142/S021812749800098X
8. Vinod Patidar, Pareek N.K., Purohit G., Sud K.K. A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption // Optics Communications. 2011. Vol. 284. No. 19. Pp. 4331–4339. DOI: 10.1016/j.optcom.2011.05.028
9. Girau B., Vlassopoulos N. Evolution of 2-dimensional cellular automata as pseudo-random number generators // Cellular automata: 10th Intern. conf. on cellular automata for research and industry: ACRI 2012 (Santorini island, Greece, Sept. 24-27, 2012): Proc. B.; Hdbl.: Springer, 2012. Pp. 611-622. DOI: 10.1007/978-3-642-33350-7_63
10. Souyah A., Faraoun K.M. An image encryption scheme combining chaos-memory cellular automata and weighted histogram // Nonlinear Dynamics. 2016. Vol. 86. No. 1. Pp. 639–653. DOI: 10.1007/s11071-016-2912-0
11. Ключарёв П.Г. Построение случайных графов, предназначенных для применения в криптографических алгоритмах, основанных на обобщенных клеточных автоматах // Математика и математическое моделирование. МГТУ им. Н.Э. Баумана. Электрон. журн. 2017. № 3. С. 77–90. DOI: 10.24108/mathm.0317.0000076
12. Деза Е.И., Деза М.М. Энциклопедический словарь расстояний: пер. с англ. М.: Наука, 2008. 444 с. [Deza E., Deza M.M. Dictionary of distances. Amst.: Elsevier, 2006. 391 p.].

Mathematics and Mathematical Modeling, 2019.
no. 03. pp. 1 – 24.

DOI: **10.24108/mathm.0319.0000181**

Mathematics & Mathematical Modelling

Electronic journal

http://mathmelpub.ru ISSN 2412-5911

A Cellular Automata Algorithm for Matrix Permutations

I.V. Matyushkin[1,*], V.S. Kozhevnikov[2]

[1]National Research University of Electronic Technology, Zelenograd, Moscow, Russia
[2]Moscow Institute of Physics and Technology, Dolgoprudny, Russia

*imatyushkin@niime.ru

Keywords: cellular automata, permutation, rearrangement, pseudorandom numbers, cryptography, metrics

Received: 06.05.2019, Revised: 21.05.2019

The article describes an algorithm to provide permutation of matrix elements through cyclic shifts of rows and columns and gives a formal description of a cellular automaton (CA) that implements this algorithm. For this, a square n × n lattice with closed boundaries and a neighbourhood of a von Neumann type cell are used.

As a result of a computational experiment for the initial orders of the matrix *n*, it was found that after a sufficiently large number of steps, the algorithm transfers the matrix to the original one, i.e. has period N. For odd orders of the matrix, the growth of N as a function of n is faster than exponential.

A movement of the individual elements of the matrix has been analysed to show that they move in a manner similar to billiard balls. The element moves to the matrix boundaries at an angle of 45 ° and changes direction when it reaches the bound. A defined explicit dependence of the element movement period on its initial position in the matrix allows us to prove that the global period N is equal to the least common multiple of all the odd numbers being less than 2n, i.e. N = LCM(3,5, ..., 2n-1).

To analyse the permutation dynamics, two authors-introduced “metrics” that reflect a degree of randomizing were used. One of the metrics is introduced specifically for the matrix, the other for the linear array and depends on how the matrix is transformed into a one-dimensional array. By searching for permutations with extreme metric values, it was found that as a result of permutations, even-order matrices undergo block permutation. In the case of an odd order, the matrix undergoes a rotation of ± 90 ° and 180 ° (reflection relative to the centre). Moreover, the direction of rotation depends on the order n. For example, for n = 5, counter-clockwise rotation occurs, and for n = 7, it is clockwise.

The algorithm can be used to generate pseudorandom numbers, and a value of its period indirectly argues for it. The period can be significantly increased through a slight complication of the algorithm, for example, by introducing different lengths for cycles of changing a direction of columns and rows. Stopping the algorithm at random time, one can consider the current permutation of the matrix as an array of random numbers or somehow transform it into one random number.

References

1. Matyushkin I.V. Algorithms of the parallel computations in the formalization of cellular automata: the sorting of strings and the multiplication of numbers by Atrubin's method. *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh system (MES)* [Problems of development of perspective micro-and nanoelectronic systems (MES], 2016, no. 4, pp. 77–81 (in Russian).
2. Matyushkin I.V., Zhemerikin A.V., Zapletina M.A. Cellular automaton algorithms for string sorting and integer multiplication by Atrubin's scheme. *Izvestiia vysshikh uchebnykh zavedenij. Elektronika* [J. "Proc. of Universities. Electronics"], 2016, vol. 21, no. 6, pp. 557–565 (in Russian).
3. Matyushkin I.V., Zapletina M.A. Data reflection and transposition in the matrix of a cellular automata computator. *Izvestiia vysshikh uchebnykh zavedenij. Elektronika* [J. "Proc. of Universities. Electronics"], 2019, vol. 24, no. 1, pp. 51–63. DOI: 10.24151/1561-5405-2019-24-1-51-63 (in Russian)
4. Ji Won Yoon, Hyoungshick Kim. An image encryption scheme with a pseudorandom permutation based on chaotic maps. *Communications in Nonlinear Science and Numerical Simulation*, 2010, vol. 15, no. 12, pp. 3998–4006. DOI: 10.1016/j.cnsns.2010.01.041
5. Weichuang Guo, Junqin Zhao, Ruisong Ye. A chaos-based pseudorandom permutation and bilateral diffusion scheme for image encryption. *Intern. J. of Image, Graphics and Signal Processin*g, 2014, vol. 6, no. 11, pp. 50–61. DOI: 10.5815/ijigsp.2014.11.07
6. Kaur M., Kumar V. Efficient image encryption method based on improved Lorenz chaotic system. *Electronics Letters*, 2018, vol. 54, no. 9, pp. 562–564. DOI: 10.1049/el.2017.4426
7. Fridrich J. Symmetric ciphers based on two-dimensional chaotic maps. *Intern. J. of Bifurcation and Chaos in Applied Sciences and Engineering*, 1998, vol. 8, no. 6, pp. 1259–1284. DOI: 10.1142/S021812749800098X
8. Vinod Patidar, Pareek N.K., Purohit G., Sud K.K. A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption. *Optics Communications*, 2011, vol. 284, no. 19, pp. 4331–4339. DOI: 10.1016/j.optcom.2011.05.028
9. Girau B., Vlassopoulos N. Evolution of 2-dimensional cellular automata as pseudo-random number generators. *Cellular Automata: 10th Intern. conf. on cellular automata for research and industry: ACRI 2012* (Santorini island, Greece, Sept. 24-27, 2012): Proc. B.; Hdbl.: Springer, 2012. Pp. 611–622. DOI: 10.1007/978-3-642-33350-7_63

10. Souyah A., Faraoun K.M. An image encryption scheme combining chaos-memory cellular automata and weighted histogram. *Nonlinear Dynamics*, 2016, vol. 86, no. 1, pp. 639–653. DOI: 10.1007/s11071-016-2912-0
11. Klyucharev P.G. Random graph construction for cryptographic applications. *Matematika i matematicheskoe modelirovanie* [Mathematics and Mathematical Modelling], 2017, no. 3, pp. 77–90. DOI: 10.24108/mathm.0317.0000076 (in Russian)
12. Deza E., Deza M.M. *Dictionary of distances*. Amst.: Elsevier, 2006. 391 p. (Russ. ed.: Deza E., Deza M.M. *Entsiklopedicheskij slovar' rasstoianij*. Moscow: Nauka Publ., 2008. 444 p.).