

© 1994 г. Ю. В. Андреев, А. С. Дмитриев

## ЗАПИСЬ И ВОССТАНОВЛЕНИЕ ИЗОБРАЖЕНИЙ В ОДНОМЕРНЫХ ДИНАМИЧЕСКИХ СИСТЕМАХ

Рассмотрено применение одномерных нелинейных динамических систем к записи и обработке изображений. Обсуждены принципы и алгоритмы записи, исследуются вопросы сжатия исходной визуальной информации. Показано, как перемежаемость и хаотическое блуждание могут быть использованы для реализации хаотического сканирования памяти на основе одномерных динамических систем.

### ВВЕДЕНИЕ

Человеческая память в отличие от адресной памяти, используемой в обычных компьютерах, организована как ассоциативная память, т. е. запись и считывание информации основаны не на местоположении ячейки памяти, а на содержании информации [1]. Существует ряд моделей, реализующих ассоциативную память, среди которых одной из самых популярных является модель нейронной сети [2—4]. Она описывается как динамическая система, а запоминаемым или распознаваемым объектам соответствуют простейшие аттракторы — устойчивые положения равновесия.

Экспериментальные данные заставляют предполагать, что не только простые, но и сложные аттракторы, в том числе хаотические, играют важную роль в обработке информации биологическими нейросистемами [5]. Поэтому представляет интерес исследование процессов обработки информации с использованием сложной нелинейной динамики.

В работах [6—8] был предложен метод записи и восстановления информации на основе устойчивых циклов одномерных отображений единичного отрезка в себя. В данной работе рассматривается применение этого метода к записи и восстановлению изображений, в том числе цветных. При этом хорошо «высвечиваются» сильные и слабые стороны рассматриваемого подхода. Например, оказывается, что наличие одинаковых фрагментов большой длины является достаточно типичной ситуацией в изображениях с большим числом элементов. Для записи таких изображений требуется значительное увеличение уровня записи, что в свою очередь ведет к резкому (экспоненциальному) увеличению среднего времени выхода итерируемого процесса на цикл, соответствующий записанному изображению.

Для увеличения емкости записи в работе вводится процедура кодирования, обеспечивающая одновременно сжатие информации и снимающая проблему «экспоненциальной катастрофы».

Рассматривается проблема создания системы ассоциативной памяти для изображений на основе одномерных динамических систем (ОДС).

Наконец, показано, как перемежаемость и хаотическое блуждание могут быть использованы для реализации памяти хаотического доступа на основе ОДС.

### 1. ЗАПИСЬ И ВОССТАНОВЛЕНИЕ ИЗОБРАЖЕНИЙ

Рассмотрим применение описанного в [6—8] метода к записи и восстановлению двумерных изображений.

Пусть имеется двумерное, в общем случае многоцветное, изображение. Первый шаг состоит в его пространственной оцифровке и переходе от исходного изображения к паттерну (оцифрованному изображению) размером  $m \times n$  ячеек. Каждая

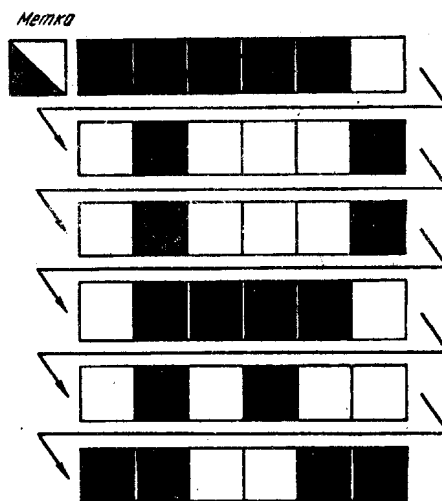
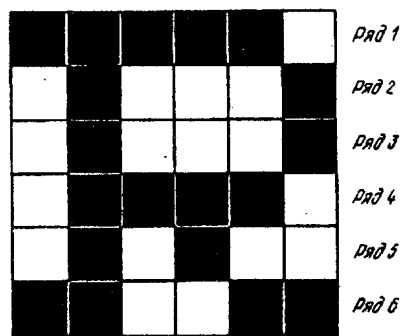


Рис. 1. Образование информационного блока из графического образа

из этих ячеек закрашивается в цвет, выбираемый из заданной палитры, и наиболее близкий к цвету, доминирующему в ячейке. Палитра из  $N_c$  цветов в рассматриваемом случае и составляет основу алфавита, а процедура закрашивания ячеек — «цветовую оцифровку» или разложение изображения по цветовым элементам алфавита. Паттерн представляется затем в виде двумерной матрицы  $A = \|a_{ij}\|$ ,  $i = 1, \dots, m$ ;  $j = 1, \dots, n$ , где  $a_{ij}$  — один из элементов алфавита.

Следующий шаг — формирование информационного блока. Для этого мы используем преобразование «телевизионного» типа, считывая паттерн (матрицу  $A$ ) построчно сверху вниз. В результате паттерн преобразуется в одномерную последовательность длины  $m \times n$ :

$$(1) \quad a_{11} \dots a_{1n} a_{21} \dots a_{2n} \dots a_{m1} \dots a_{mn}$$

Отметим, что устойчивый цикл ОДС, соответствующий информационному блоку, соответствует и всем его циклическим перестановкам. Можно выразиться и так, что в переходном процессе выхода на цикл система теряет начальную фазу.

Подобная ситуация нежелательна при работе с изображениями, поскольку операция обратного преобразования цикла в паттерн в этом случае неоднозначна.

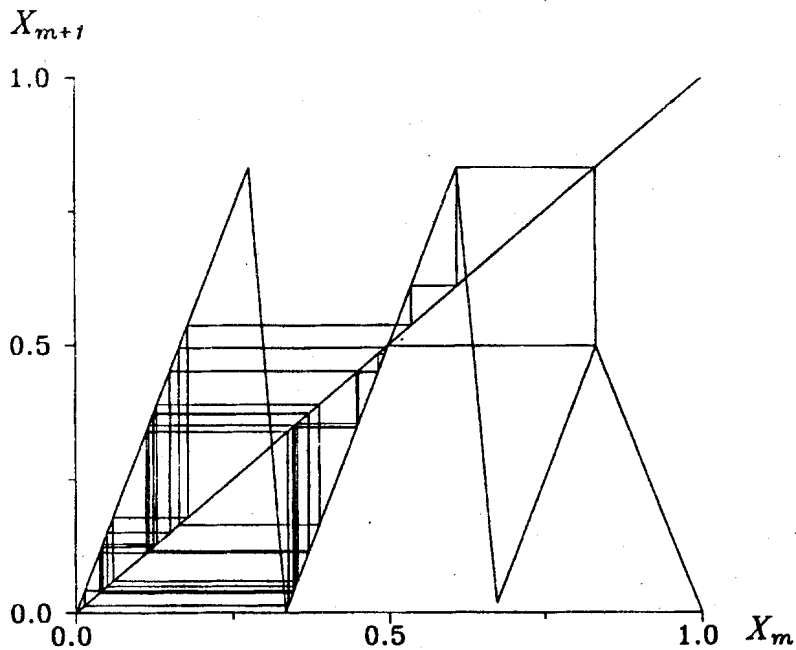
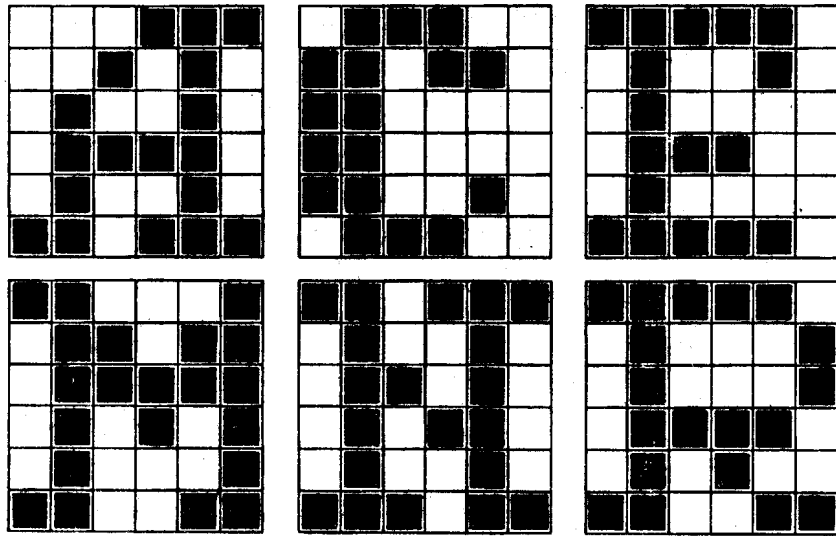


Рис. 2. Запись графических образов. *a* — Набор паттернов, записанных на двенадцатом уровне. *b* — Отображение с записанными паттернами. Показан предельный цикл, соответствующий изображению буквы *R*.

Поэтому начало информационного блока помечается специальным символом (меткой). Соответственно при восстановлении изображения по значениям переменной на предельном цикле началом изображения считается первый элемент после метки (сама метка не изображается). В качестве метки используется специальный элемент алфавита.

Рассмотрим применение описанной процедуры к задаче записи и восстановления бинарных изображений.

Пусть требуется записать на одномерное отображение букву *R*, изображенную на рис. 1. Представим ее в виде последовательности бинарных элементов (черных

и белых) и добавим в начало этой последовательности элемент, соответствующий метке. Для удобства можно считать, что алфавит состоит из трех элементов 0, 1, 2, где 0 соответствует черному цвету, 1 — белому и 2 — метке. Тогда информационный блок может быть представлен в виде

(2) 2111110010001010001011110010100110011.

Теперь нужно определить минимальный уровень, на котором можно произвести запись. В работе [6] вводится понятие записи информации на уровне  $q$ , означающего в некотором смысле введение связи между соседними элементами информационного блока. При этом построение каждой точки цикла происходит по фрагменту информационного блока длины  $q$ , а не по отдельному элементу информационного блока. Из способа построения отображения следует [6], что при записи на уровне  $q$  информационные блоки не должны содержать фрагментов, длиннее  $q - 1$ . Из (2) видно, что эта последовательность содержит два идентичных фрагмента (11110010) длины 8, начинающихся с позиций 3 и 21. Следовательно, во всяком случае уровень записи не может быть  $< 9$ .

Подобным же образом можно разложить в информационные блоки и другие буквы из приведенных на рис. 2, а. Отображение с этими записанными блоками информации приведено на рис. 2, б (уровень записи — 12). В качестве примера показан цикл для паттерна буквы R. Интересно отметить, что по своей «грубой» структуре отображение близко к отображению для сдвига Бернулли из трех символов.

Теперь рассмотрим применение предложенного подхода к записи и восстановлению цветных изображений. В этом случае алфавит будет состоять из  $N_c + 1$  элементов, где  $N_c$  — число цветов в паттерне. Основная трудность при записи таких изображений связана с возможным присутствием в паттерне достаточно длинных одинаковых фрагментов. Например, для паттерна на рис. 3, а максимальная длина двух одинаковых фрагментов равна 30. Поскольку длина информативного отрезка отображения пропорциональна  $(N_c + 1)^2$ , наличие в рисунке одинаковых фрагментов большой длины приводит к необходимости использования при записи вычислений с двойной точностью.

Кроме того, среднее время сходимости процесса к циклу при произвольных начальных условиях быстро растет с увеличением уровня записи.

## 2. ОЦЕНКА ИНФОРМАЦИОННОЙ ЕМКОСТИ ЗАПИСИ НА ОСНОВЕ ОДНОМЕРНОГО ОТОБРАЖЕНИЯ

Вопрос об информационной емкости записи на одномерном отображении является естественным при анализе возможностей метода и его конкурентоспособности по сравнению с другими методами записи и восстановления информации.

Очевидно, что при записи информации с помощью рассматриваемого метода существуют ограничения. Например, пусть алфавитом является десятичная система счисления. Блок информации 174 можно записать на первом уровне. Одновременно с ним можно записать и блок информации 268. Однако запись трех блоков информации 174, 268 и 173 на одном отображении невозможна, так как отображение при этом становится неоднозначным. Вместе с тем, если запись вести на третьем уровне или выше, то все три блока информации могут быть записаны на одном отображении.

Из способа построения одномерного отображения, описанного в [6—8], следует основное условие, определяющее возможность записи образов: отсутствие одинаковых фрагментов длины, равной или превышающей уровень записи. Если оно не выполняется, отображение теряет однозначность. Это и есть условие «ортогональности» образов для данной системы записи и хранения информации.

Для ответа на вопрос об информационной емкости метода необходимо выяснить,

сколько существует «ортогональных» образов длины  $n$  из символов алфавита длины  $N$  при записи на уровне  $q$ .

Рассмотрим запись образов без меток.

Сначала оценим, сколько вообще существует последовательностей длины  $n$ . Отметим, что цикл для информационного блока совпадает с циклами для информационных блоков, представляющих все циклические перестановки исходного информационного блока (всего их  $n$ ). Поэтому, например, последовательности 174, 741, 417 при данном способе записи неразличимы и представляют один и тот же информационный блок. Поскольку существуют последовательности, совпадающие с некоторыми из своих циклических перестановок (например последовательность, состоящая из  $n$  повторений одного символа алфавита), то полное число разных последовательностей длины  $n$  равно или больше  $N^n/n$ .

Рассмотрим случай  $q \geq n$ , т. е. когда уровень записи больше длины образа. Пусть  $p$  — число информационных блоков, совпадающих с некоторыми из своих циклических перестановок. Эти информационные блоки не могут быть записаны на одномерном отображении как последовательности длины  $n$ . Все информационные блоки, за исключением этих, в принципе, могут быть записаны, так как две различные последовательности длины  $n$  могут иметь идентичные фрагменты только длины, меньшей  $n$ , и следовательно, меньшей  $q$ . Значит, эти последовательности удовлетворяют условию «ортогональности». Полное число информационных блоков, которые могут быть записаны в этом случае, поэтому равно  $(N^n - p) / n \leq N^n/n$ .

Во втором случае при  $q < n$  уровень записи меньше длины образа. Пусть записывается информационный блок  $C = c_1 c_2 \dots c_n$ , где  $c_i$  — элементы алфавита, на уровне записи  $q$ ,  $q < n$ . Чтобы проверить, удовлетворяет ли  $C$  условию «ортогональности», необходимо рассмотреть все фрагменты этого информационного блока, имеющие длину  $q$ ,  $C_i = c_i c_{i+1} \dots c_{i+q-1}$ ,  $i = 1, \dots, n$ . Если все эти фрагменты различны, то информационный блок может быть записан. Однако эти фрагменты уже не могут быть использованы ни в каком другом записываемом образе.

Итак, полное число возможных фрагментов длины  $q$  в рассматриваемом случае равно  $N^q$ . При записи одного информационного блока используются  $n$  из них, все они различны из-за условия «ортогональности». Остальные блоки, содержащие эти фрагменты, уже не могут быть записаны. Записанный информационный блок длины  $n$  содержит  $n$  различных фрагментов длины  $q$ . Поэтому из всех возможных информационных блоков длины  $n$  (а таких насчитывается  $\geq N^n/n$ ) можно записать максимум  $N^q/n$  блоков.

Итак, информационная емкость памяти при записи на устойчивых циклах одномерных отображений

$$(3) \quad E \leq \begin{cases} N^n/n, & n \leq q, \\ N^q/n, & n \geq q. \end{cases}$$

Приведем пример. Выясним, сколько слов из шести букв русского алфавита можно записать на третьем уровне. Алфавит состоит из 33 букв (элементов). Полное число информационных блоков (слов) с учетом циклических перестановок, включая экзотические «бббббб» или «ыыыытт» больше, чем  $33^6/6$ . Если записано слово «прибор», то нельзя записать блоки, содержащие фрагменты «при», «риб», «ибо», «бор», «орп», «рпр», а записав одновременно со словом «прибор» слово «любовь», мы исключаем из списка доступных в дальнейшем фрагментов фрагменты «люб», «юбо», «бов», «овь», «вьл», «блю». При записи шестибуквенных слов на третьем уровне ( $N = 33$ ,  $n = 6$ ,  $q = 3$ ) емкость  $E \leq 5989$ .

Из формулы (3) следует, что в предельном случае  $E = 1$ ,  $n = N^q$ , т. е. для  $n > N^q$  невозможно записать ни одного образа. С другой стороны, для выбранных  $N$  и  $n$  минимальный уровень записи  $q \geq \log_n N$ .

Представляет интерес также относительная емкость данного метода записи и

хранения информации, т. е. соотношение числа возможных и записанных образов. Нетрудно видеть, что

$$(4) \quad E_{\text{отн}} \leq \begin{cases} 1, & n \leq q = f(N, n - q) \\ 1/N^{n-q}, & n \geq q \end{cases}$$

Из формулы (4) следует, что относительная емкость зависит только от разности  $n$  и  $q$  и по мере уменьшения уровня записи относительно длины образа экспоненциально падает.

### 3. «ОРТОГОНАЛИЗАЦИЯ» ОБРАЗОВ И СЖАТИЕ ИНФОРМАЦИИ

В разд. 1 показано, что на уровне  $q$  можно записать любой информационный блок, в котором отсутствуют одинаковые фрагменты длины  $L \geq q$ . Если же такие фрагменты встречаются, то запись невозможна, так как функция отображения становится неоднозначной. Это ограничение можно обойти, увеличив уровень записи. Однако на этом пути имеются два принципиальных ограничения. Первое из них заключается в том, что при превышении некоторого уровня  $q_{кр}$  мы вынуждены будем перейти от вычислений с одинарной точностью к вычислениям с двойной точностью. Это связано с тем, что размеры информативных отрезков становятся сравнимыми с точностью вычислений. Второе ограничение несколько другого плана. Суммарная мера информативных участков при высоком уровне записи может оказаться весьма малой по отношению к единице. При этом мы сталкиваемся с ситуацией когда при задании произвольных начальных условий траектория в среднем долго или очень долго блуждает по фазовому пространству, прежде чем попадет в окрестность устойчивого предельного цикла.

Парадокс заключается в том, что наибольшие трудности при записи информации рассмотренным в предыдущем разделе методом возникают при записи информационных блоков, содержащих однородные фрагменты. Из общих соображений ясно, что они содержат весьма ограниченное количество информации, и в то же время именно с ними связаны основные трудности с ее записью. Создается впечатление, что используемый способ записи информации не слишком эффективен для таких ситуаций.

Ниже вводится способ «ортогонализации» информационных блоков, позволяющий снять эти ограничения и одновременно осуществить сжатие информации.

Будем действовать следующим образом. Если при просмотре информационного блока (или совокупности блоков), который надо записать на уровне  $q$  с помощью алфавита из  $N$  символов, обнаруживаются два идентичных фрагмента длины  $q$ , то в алфавит вводится новый элемент, представляющий собой этот фрагмент, и все такие фрагменты (их может оказаться больше двух) в информационном блоке (блоках) заменяются на новый элемент алфавита.

После этого снова производится просмотр информационного блока (совокупности информационных блоков). Если в нем снова появляются, теперь уже при алфавите длины  $N + 1$ , одинаковые фрагменты длины  $q$ , то вводится следующий новый элемент и т. д. Иными словами, мы фиксируем  $q$ , увеличивая  $N$  (и уменьшая  $n$ ) до тех пор, пока информационный блок не сможет быть записан.

Такая процедура позволяет записать произвольный информационный блок на любом уровне записи, начиная со второго.

Приведем простой пример. Пусть для записи черно-белых изображений имеется алфавит, состоящий из двух элементов (черного и белого). Запишем изображение черного квадрата, разбитое сеткой размером  $8 \times 8$  элементов, т. е. информационный блок размером 64, на втором уровне записи, используя введенный принцип кодирования. Нетрудно видеть, что фрагмент, состоящий из первых двух элементов

Уровень за- писи	Рис. 3, а (16 × 24)		Рис. 3, б (32 × 48)		Рис. 3, в (64 × 96)	
	Длина	Алфавит	Длина	Алфавит	Длина	Алфавит
2	81	35	207	88	453	190
3	103	25	249	54	569	136
4	103	26	301	47	607	117

блока, представляет собой новый элемент алфавита, первые четыре элемента также представляют собой новый элемент алфавита и т. д.

Обозначим элементы исходного алфавита  $w$  (белый) и  $b$  (черный), дополнительные элементы  $b_2$  (третий элемент),  $b_3$  (четвертый) и т. д. Информационный блок кодируется в новом алфавите как цикл периода 1, состоящий из элемента  $b_7$ . Для изображения черного квадрата размером  $64 \times 64$  новый алфавит состоит из белого элемента ( $w$ ), черного элемента ( $b$ ), черной пары ( $b_2 = bb$ ),  $b_3 = b_2b_2 = bbbb$ ,  $b_4, \dots, b_{13}$ . Весь новый алфавит будет содержать 15 элементов, а информационный блок будет описываться циклом периода 1, состоящим из пятнадцатого элемента алфавита.

Таким образом, «ортогонализация» сопровождается сжатием информации.

Конечно, приведенная в примере эффективность сжатия информации представляет собой вырожденный случай. Чтобы оценить реально достижимое сжатие, рассмотрим результаты применения кодирования к цветному изображению. На рис. 3, а—в представлены результаты редукции исходного изображения к паттернам размером  $16 \times 24$ ,  $32 \times 48$ ,  $64 \times 96$  элементов. Всего в изображении имеется семь цветов, поэтому исходный алфавит состоит из восьми элементов (семь цветов плюс метка начала изображения). В таблице приведены данные о числе элементов алфавита и длине описывающего паттерн цикла для различных уровней записи. Из нее следует, что во всех рассматриваемых случаях удастся произвести запись, используя вычисления с одинарной точностью. Коэффициент обратимого сжатия при этом находится в пределах от 3 до 10.

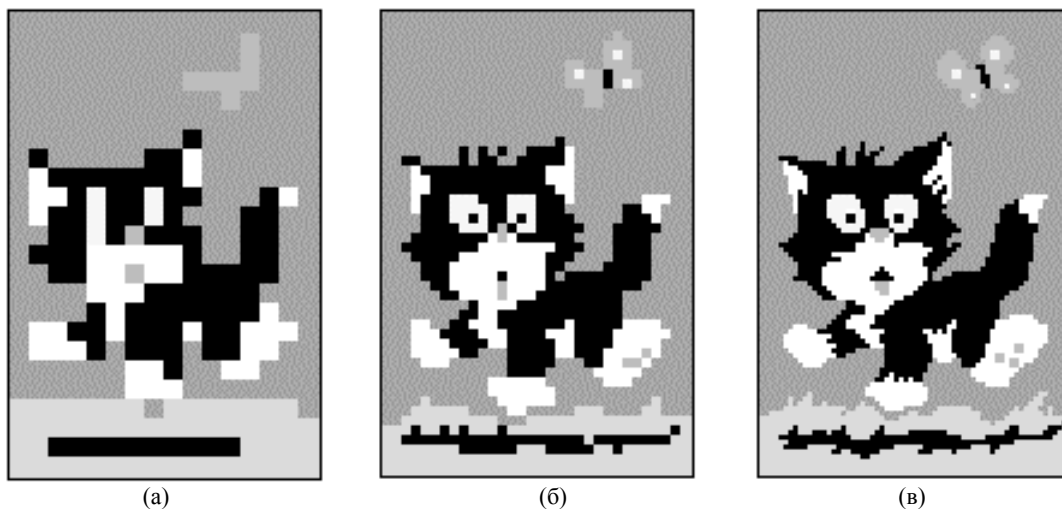


Рис. 3 Редукция изображения котенка к паттернам размером:  $16 \times 24$  (а),  $32 \times 48$  (б),  $64 \times 96$  (в)

#### 4. ХАОТИЧЕСКОЕ СКАНИРОВАНИЕ ПАМЯТИ НА ОСНОВЕ ОДНОМЕРНОГО ОТОБРАЖЕНИЯ

В рассмотренном методе записи и восстановления информации на основе одномерного отображения для записи информационных блоков использовались устойчивые предельные циклы. В этом разделе будет показано, что можно организовать хаотический доступ в одномерной памяти, в том смысле, что траектория будет посещать все области, соответствующие информации, записанной на отображении. Очевидно, что использовать устойчивые предельные циклы для этой цели нельзя, так как траектория любой начальной точки в бассейне притяжения устойчивого цикла притянется к этому циклу.

Для организации хаотического сканирования памяти можно использовать явление перемежаемости в динамических системах. Хаотическое сканирование может быть реализовано с использованием неустойчивых циклов для записи информационных блоков. Каждый цикл легко может быть сделан неустойчивым. Для этого произведение тангенсов углов наклона функции отображения в

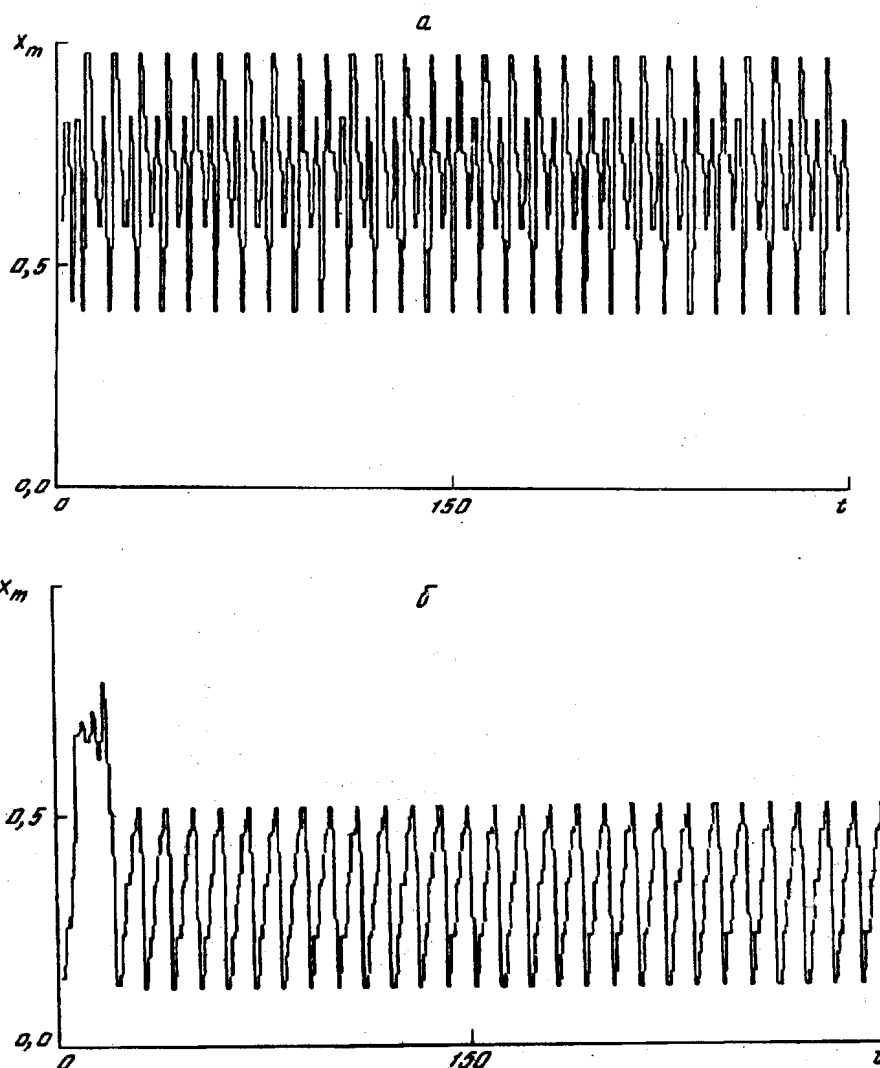


Рис. 4, а, б

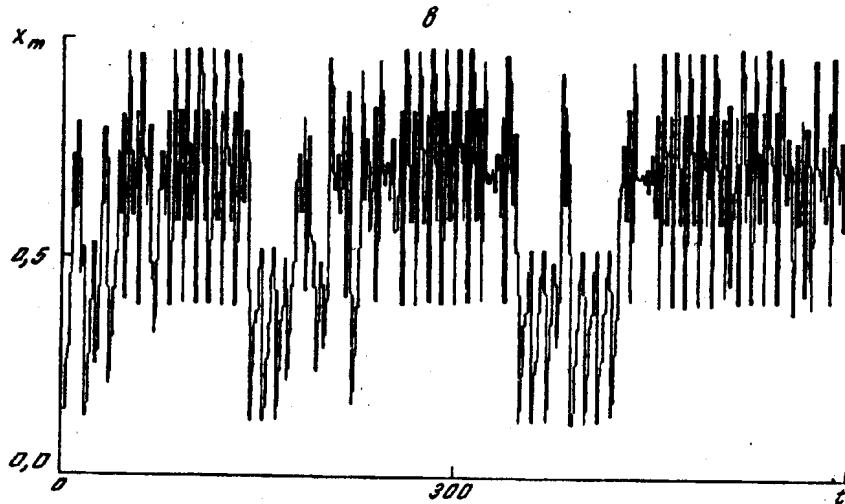


Рис. 4. Временные реализации траектории одномерного отображения с записанными на втором уровне информационными блоками 97583 и 12345. Сходимость к устойчивым циклам, соответствующим блокам 97583 (а), 12345 (б), перемежаемость между двумя циклами (в)

точках цикла необходимо положить по модулю немного больше единицы. Траектории, начинающиеся в окрестности цикла, будут в этом случае покидать его через некоторое время и блуждать по фазовому пространству, пока не попадут в окрестность этого или другого предельного цикла с такой же неустойчивостью. Траектория будет «застывать» в окрестности этого предельного цикла на некоторое время, прежде чем покинуть его снова. Если независимо от начальных условий траектория посещает окрестности всех точек, то будет наблюдаться перемежаемость по отношению ко всем циклам, соответствующим записанной информации.

Если наклоны информационных участков слишком велики, траектории будут покидать окрестности предельных циклов, соответствующих записанной информации, очень быстро. Могут возникнуть и другие динамические явления. Анализ этих вопросов требует дальнейшего исследования.

Здесь мы ограничимся следующими двумя примерами.

На рис. 4 показан простой пример перемежаемости между двумя циклами, соответствующими информационным блокам 97583 и 12345. Алфавит образуют цифры 0,1,...,9, используется второй уровень записи. Если наклоны информационных участков выбраны  $< 1$ , то траектория одномерного отображения сходится к одному из двух предельных циклов (рис. 4, а, б). Однако если сделать наклоны немного  $> 1$ , то будет наблюдаться перемежаемость (рис. 4, в). В этом случае система «блуждает» между двумя циклами.

Покажем возможность создания памяти с хаотическим доступом на более сложном примере. Запишем 10 информационных блоков, соответствующих десяти паттернам размером  $8 \times 8$  элементов, используя 2 цвета (3 элемента в исходном алфавите). Записанная информация состоит из образов букв латинского алфавита A, b, C, d, E, f, G, h, I, j (рис. 5, а). Информационные блоки записаны на втором уровне. Метод построения отображения такой же, как и при записи на устойчивых циклах, за исключением того, что наклоны информативных участков выбраны немного  $> 1$ . Последовательность моментальных снимков (рис. 5) демонстрирует перемежающееся появление всех десяти образов.

Траектория посещает все участки с записанной информацией. Переключая наклон информативных участков (делая их по модулю  $< 1$ ), можно сделать неустойчивый цикл устойчивым. Траектория при этом сойдется к некоторому

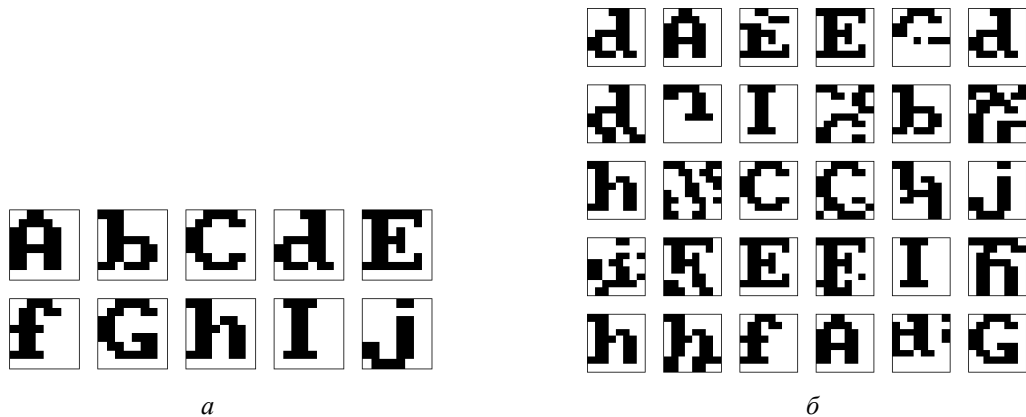


Рис. 5. Перемежаемость между паттернами, представляющими буквы. Паттерны, записанные на втором уровне (а). Последовательность моментальных снимков, иллюстрирующая перемежаемость (б). Изображения фиксируются через произвольные моменты времени и показываются в хронологическом порядке слева направо

циклу, восстанавливая таким образом тот информационный блок, в бассейне притяжения которого находилась траектория в момент переключения.

#### ЗАКЛЮЧЕНИЕ

Рассмотренный в работе метод записи изображений на предельных циклах ОДС и извлечения изображений, записанных на циклах, существенным образом использует ОДС как среду для записи, хранения и восстановления информации. Простота используемых динамических систем позволяет исследовать с их помощью ряд принципиальных вопросов, связанных с общими принципами обработки информации в системах со сложной динамикой. Некоторые из этих вопросов обсуждены в статье.

Вместе с тем показано, что эти системы обладают достаточно высокой информационной емкостью, что позволяет надеяться на возможность создания программных и аппаратных средств, использующих одномерные кусочно-линейные отображения для решения прикладных задач.

#### СПИСОК ЛИТЕРАТУРЫ

1. Kohonen T. Content-Addressable Memories, B.: Springer-Verlag, 1980.
2. Hopfield J. J.//Proc. Natl. Acad. Sci. 1982. V. 79. P. 2554.
3. Grossberg S.//Neural Networks. 1988. V. 1. P. 17.
4. Carpenter G. A.//Neural Networks. 1989. V. 2. P. 243.
5. Skarda C. A., Freeman W. J.//Behavioral and Brain Sci. 1987. V. 10. P. 161.
6. Дмитриев А. С.//РЭ. 1991. Т. 36. № 1. С. 101.
7. Dmitriev A. S., Panas A. I., Starkov S. O.//Phys. Lett. A. 1991. V. 155. № 8, 9. P. 494.
8. Андреев Ю. В., Бельский Ю. Л., Дмитриев А. С.//Докл. междунар. семинара «Нелинейные цепи и системы». М., 1992. Т. 1. С. 51.

Поступила в редакцию  
11.01.1993 г.