2008 Математические методы криптографии № 2(2)

DOI 10.17223/20710410/2/9

УДК 62-50:519.7

ОБРАТИМЫЕ ДИНАМИЧЕСКИЕ СИСТЕМЫ С ПЕРЕМЕННОЙ РАЗМЕРНОСТЬЮ ФАЗОВОГО ПРОСТРАНСТВА В ЗАДАЧАХ КРИПТОГРАФИЧЕСКОГО ПРЕОБРАЗОВАНИЯ ИНФОРМАЦИИ

А.М. Ковалев, В.А. Козловский, В.Ф. Щербак

Институт прикладной математики и механики НАН Украины, г. Донецк

E-mail: {kovalev, kozlovskii, shvf}@iamm.ac.donetsk.ua

Рассматривается способ преобразования информации, основанный на применении теории обратимых систем управления. Предложен метод изменения размерности пространства состояний, усложняющий поведение системы без изменения структуры уравнений. Тем самым вводятся структурные ключи, которые также могут быть секретными.

Ключевые слова: *обратные системы управления*, *защита информации*, *автомат*, *инвариантное множество.*

В последнее время различные аспекты теории нелинейных динамических систем со сложным (хаотическим) поведением траекторий находят применение в области обработки и защиты информации. В данной работе реализован подход, связанный с использованием методов теории обратимых систем управления в коммуникационных технологиях [1, 2]. В работе рассматривается метод преобразования и передачи информации, использующий дуальное нелинейное преобразование информации по следующей схеме: оцифрованное сообщение подается как внешнее воздействие на вход динамической системы, информация о ее траекториях, неявно зависящая от входа, в виде сигнала направляется в коммуникационные сети. В случае наличия у передающей системы свойства обратимости может быть синтезирована обратная система, играющая роль дешифратора (приемник). В работе показано, что при использовании обратных систем может возникнуть эффект, названный динамической деградацией. Он связан с возможностью попадания траекторий на инвариантные множества в фазовом пространстве, лежащие на многообразиях меньшей размерности. В общем случае это оказывает негативное влияние на сложность поведения и, таким образом, уменьшает стойкость соответствующих алгоритмов шифрования к определенным видам атак. Поэтому представляет интерес задача нахождения способов компенсации таких вырождений динамики. Для дискретных систем в рамках этой схемы предложен метод управления размерностью пространства состояний и/или входов.

1. Обратимые динамические системы

Предположим, что передатчик является дискретной динамической системой, правые части которой зависят от вектора функции $u(.)$ – оцифрованного информационного сообщения:

$$x(t+1) = f(x(t), u(t)), x(0) = x_0; \quad (1)$$

$$y(t) = h(x(t), u(t)), \quad (2)$$

где $x(t) \in \mathrm{R}^n$, $u(t) \in \mathrm{R}^m$, $y(t) \in \mathrm{R}^p$ определяют векторы состояния системы, ее вход и выход соответственно. По каналам связи передается выходной сигнал – функция $y(t)$, зависящая от состояния системы, ее параметров и сообщения $u(t)$. Для построения уравнений, описывающих динамику приемного устройства, рассмотрим задачу восстановления значений входного воздействия по значениям функции выхода. В теории управления непрерывными динамическими системами одним из способов ее решения является построение системы, обратной к исходной [2, 3]. В отличие от известных схем преобразования информации (гаммирования, маскирования сигнала), входная информационная последовательность $u(t)$ подается на вход динамической системы, и выход $y(t)$ при этом может и не зависеть явно от $u(t)$. При этом система (1), (2) порождает однозначное отображение вход – выход

$$\{x(0), u(0), u(1), \ldots\} \to \{y(0), y(1), \ldots\} \quad (3)$$

по формулам

$$y(0) = h(x(0), u(0)) = h_0(x(0), u(0)),$$

$$y(1) = h(x(1), u(1)) = h(f(x(0), u(0)), u(1)) = h_1(x(0), u(0), u(1)),$$

$$y(2) = h(x(2), u(2)) = h(f(x(1), u(1)), u(2)) = h_2(x(0), u(0), u(1)),$$

$$y(t) = h(x(t), u(t)) = \ldots = h_t(x(0), u(0), \ldots, u(t)), \quad (4)$$

при котором неизвестному начальному состоянию и значениям последовательности $\{u(0), u(1), ...\}$ соответствует известная выходная последовательность $\{y(0), y(1), ...\}$. Многие теоретические и практические задачи теории управления, связанные с определением состояния и параметров системы (1), построением обратных связей, сводятся к обращению этого отображения. Один из способов такого обращения может быть реализован с помощью обратной системы, т.е. системы вход – выход, у которой входом служит информация об $y(.)$ на некотором интервале, а выходом является функция $u(.)$.

Введем понятие относительного порядка входа для системы (1), (2). Так как значение функции $h(x(t), u(t))$ может не зависеть явно от значений $u(t)$, то аналогично и правая часть выражения $y(t+1) = h(f(x(t), u(t))$ может не содержать всех компонент вектора $u(t)$. Определим, на сколько шагов происходит задержка между информацией на входе и выходе системы (1), (2). Это величина и указывает на относительный порядок входа. Пусть $Y_t = (y(0), y(1), ..., y(t))$, $H_t = (h_0(.), h_1(.), ..., h_t(.))$, $x = x(0)$, $u = u(0)$, остальные компоненты векторов входной последовательности, содержащиеся в правых частях (4), обозначим $v_t = (u(1), u(2), ..., u(t))$. В этих обозначениях передаточное отображение (3) может быть переписано в виде

$$Y_t = H_t(x, u, v_t). \tag{5}$$

Будем говорить, что система (1), (2) имеет относительный порядок $\alpha > 0$ в некоторой области, если для всех x, u, v_t из этой области

$$\operatorname{rank}\frac{\partial H_i(x,u,v_i)}{\partial u} < m, \quad i = 0, 1, ..., \alpha-1; \quad \operatorname{rank}\frac{\partial H_\alpha(x,u,v_\alpha)}{\partial u} \equiv m.$$

Таким образом, относительный порядок α для дискретной системы указывает на номер элемента выходной последовательности, на которое явно влияют все компоненты первого элемента входной последовательности – вектора $u(0)$. В общем случае решение уравнений $Y_\alpha = H_\alpha(x, u, v_\alpha)$ относительно u имеет вид $u = H_\alpha^{-1}(Y_\alpha, x, v_\alpha) = G(Y_\alpha, x, v_\alpha)$, что не позволяет определить u без знания значений v_α. Достаточным условием того, что решение алгебраической системы (5) не зависит от v_α , является равенство [2]

$$\operatorname{rank}\frac{\partial H_i(x,u,v_\alpha)}{\partial(u,v_\alpha)} = m + \operatorname{rank}\frac{\partial H_i(x,u,v_\alpha)}{\partial v_\alpha}.$$

Подставляя выражение $u = G(Y_\alpha, x)$ в уравнения (1), получаем динамическую систему

$$x(t+1) = f(x(t), G(Y_\alpha(t), x(t))), \quad x(0) = x_0, \tag{6}$$

выход которой совпадает со входом исходной системы (1)

$$u(t) = G(Y_\alpha(t), x(t)). \tag{7}$$

Система (6), (7) является обратной динамической системой управления. Так как, по построению, $Y_\alpha(t) = (y(t), y(t+1), ..., y(t+\alpha))$, то в обратной системе в качестве входа должен присутствовать фрагмент будущих значений выхода (2). При одинаковых начальных состояниях траектории исходной и обратной систем совпадают. Поэтому можно считать, что обратная система является альтернативной формой описания одного и того же отображения вход – выход (4).

Для систем со скалярным входом и выходом ($m = p = 1$) условия однозначного обращения отображения (4) заметно упрощаются. Равенства (5) в этом случае не содержат переменных v_α. При этом относительный порядок означает номер элемента выходной последовательности, на который явно влияет первый элемент входной последовательности.

Пример. Рассмотрим передатчик (шифратор) – нелинейную систему вход – выход, на вход которой подается сообщение $u(t)$

$$\begin{aligned} x_1(t+1) &= x_2(t)x_3(t) \bmod N, \\ x_2(t+1) &= x_1(t)x_3(t) + u(t) \bmod N, \\ x_3(t+1) &= x_1(t)x_2(t) \bmod N, \\ y(t) &= x_2(t). \end{aligned} \tag{8}$$

Сигнал $y(t)$ направляется в коммуникационную сеть. Ключом для расшифрования являются неизвестные начальные условия системы $x_1(0)$ и $x_3(0)$. Приемник (дешифратор) – обратная система, с помощью которой при известном ключе проводится восстановление состояния передающей системы

$$\begin{aligned} X_1(t+1) &= X_2(t)X_3(t) \bmod N, X_1(0) = x_1(0), \\ X_2(t+1) &= y(t+1), \\ X_3(t+1) &= X_2(t)X_1(t) \bmod N, X_3(0) = x_3(0). \end{aligned} \tag{9}$$

Искомое входное воздействие определяется по формуле

$$u(t) = y(t+1) - X_1(t)X_3(t) \bmod N. \tag{10}$$

2. Эффект динамической деградации

Система (8) в отсутствие входного воздействия является нелинейной, и ее траектории при больших N обладают достаточно сложным поведением. Исключением являются траектории, лежащие на инвариантных множествах: $x_i(.) = 0$ либо $x_i(.) = x_j(.)$, $i, j = 1, 2, 3$. В первом случае состояние системы уже через шаг переходит в положение равновесия – начало координат. В общем случае, траектории, попав на инвариантные множества, остаются на нем во все последующие моменты, что приводит к падению размерности фазового пространства состояний системы. При введении в правую часть последовательности $u(t)$ система (8) становится неавтономной, и естественно предполагать, что это лишь усложнит динамику выхода и тем самым повысит сложность задачи восстановления входа. Вместе с тем анализ результатов моделирования процесса передачи информации с помощью описанной схемы показывает, что, начиная с некоторого момента t, выход системы в точности совпадает с ее входом в момент t–1. Из этого следует предположение, что собственная динамика системы (8) перестает влиять на передаваемый сигнал.

Действительно, неавтономная система (8) также обладает инвариантными множествами $x_1(.) = 0$, $x_3(.) = 0$ и $x_1(.) = x_3(.)$. При этом из полученных формул следует следующее рекуррентное выражение для определения $u(t)$:

$$u(t) = y(t + 1) - C \prod_{j=0}^{t} y(j),$$

где C равно $x_1(0)$ для четных и $x_3(0)$ для нечетных значений t. Из последнего равенства, в частности, следует, что если для некоторого целого M значение выхода $x_1(M)\, x_3(M) + u(M) = 0 \bmod N$, то для любого $i > M$ имеем $y(i) = u(i–1)$. Таким образом, вместо шифрования информационной последовательности $u(i)$ выход рассматриваемой динамической системы для любых значений ключевых параметров, начиная с некоторого момента, в точности передает значение входа с единичной задержкой.

Определение. Эффект вырождения собственной динамики системы при введении в правую часть неавтономного возмущения, выраженный в виде падения размерности пространства состояний, назовем динамической деградацией.

Безусловно, при составлении схем преобразования и передачи информации с использованием динамических систем необходим учет этого эффекта. Для того чтобы избежать влияния динамической деградации, можно применить следующую схему. Поскольку динамика состояний для передающей и принимающей систем совпадает, то достаточно рассмотреть одну из них, например систему (1), (2). На первом шаге, одновременно с выбором уравнений для передатчика, требуется найти явное описание всех инвариантных множеств, допускаемых этой системой. Далее, при разработке алгоритмов передачи и приема сигнала должна быть предусмотрена проверка условий вырождения (попадания траектории на инвариантное множество). В случае такого попадания в некоторый момент t передающая и принимающая системы должны по согласованному правилу изменить последующее состояние на состояние, не принадлежащее инвариантному множеству. Тем самым траектория будет выведена, по крайней мере на какое-то число шагов, за его пределы. В частности, для рассмотренного примера может быть применено следующее правило: при наступлении в момент t одного из событий $x_i(t) = 0$, $i \in \{1, 2, 3\}$ или $x_1(t) = x_3(t)$ на следующем шаге системы (8), (9) стартуют с исходного начального условия: $x(t+1) = X(t+1) = x_0$.

3. Автоматы-аналоги

При компьютерном моделировании бесконечных динамических систем фактически осуществляется замена исходной системы некоторым конечным аналогом, сохраняющим требуемые свойства исходной системы с определенной точностью. Оставляя в стороне вопросы точности моделирования исходной системы конечной, будем рассматривать заданную систему как прототип для построения конечной системы-аналога, переход к которой может быть осуществлен различными способами. В работе выбран один из возможных вариантов такой замены исходной динамической системы конечным автоматом с достаточно большими входным, внутренним и выходным алфавитами. При этом автомат описывается системой уравнений в конечном поле или кольце, возникающих естественным образом из описания исходной системы.

Автомат понимается как пятерка объектов $A = (X, U, Y, \delta, \lambda)$ [4], где X – множество состояний, U – входной алфавит, Y – выходной алфавит, δ: $X \times U \to X$ – функция переходов, λ: $X \times U \to Y$ – функция выходов.

Автомат называется автоматом без потери информации (БПИ), если из равенства $\lambda(x, u_1) = \lambda(x, u_2)$ следует равенство $u_1 = u_2$ для любых $x \in X$, $u_1 \in U$, $u_2 \in U$. Если множества состояний, входов и выходов автомата конечны, автомат называется конечным. Из контекста будет понятно, когда рассматриваются конечные автоматы. Функции δ и λ расширяются на множество U^* слов конечной длины обычным образом.

Далее автомат удобно описывать системой уравнений над конечным кольцом или полем. В этом случае его функционирование рассматривается в дискретном времени $t \in T = \{0, 1, 2, ...,\}$ и задается каноническими уравнениями, например, в таком варианте:

$$x(t+1) = \delta(x(t), u(t)),$$
$$y(t) = \lambda(x(t), u(t)), \; t \in T. \qquad (11)$$

Оставляя в стороне вопросы приближения, возникающие при переходе от исходной системы к ее автоматной модели, принимаем во внимание лишь то, что конечность числа значений участвующих в них величин и необходимость сохранения формы уравнений, отражающей связи между этими величинами, делают естественным рассмотрение этих уравнений как уравнений в конечных полях [5] или кольцах. Поле, содержащее q элементов, обозначается через $GF(q)$. Ниже уравнения (12) дают пример такого автомата (названного автоматом Лоренца [6]), получающийся в результате перехода от непрерывной системы Лоренца как прототипа [7] и введения в нее входного воздействия:

$$\begin{aligned} x_1(t+1) &= x_1(t) + hA_1(x_2(t) - x_1(t)), \\ x_2(t+1) &= x_2(t) + h(A_2x_1(t) - x_2(t) - x_1(t)x_3(t) + Au(t)), \\ x_3(t+1) &= x_3(t) + h(x_1(t)x_2(t) - A_3x_3(t)) \\ y(t) &= x_2(t) + h(A_2x_1(t) - x_2(t) - x_1(t)x_3(t) + Au(t)). \end{aligned} \tag{12}$$

Уравнения (13) описывают обратный автомат:

$$\begin{aligned} X_1(t+1) &= X_1(t) + hA_1X_2(t) - X_1(t)), \\ X_2(t+1) &= y(t+1), \\ X_3(t+1) &= X_3(t) + h(X_1(t)X_2(t) - A_3X_3(t)), \\ u(t) &= (((y(t) - X_2(t))\cdot h^{-1} - A_2X_1(t) + X_2(t) + X_1(t)X_3(t))\cdot A^{-1}. \end{aligned} \tag{13}$$

Далее все операции в уравнениях понимаются как операции в некотором поле $GF(q)$. Заметим, что входной сигнал в систему можно вводить разными способами, выполняя лишь требование обратимости системы. В автоматном случае это означает, что автомат-преобразователь должен быть БПИ-автоматом [4] или без потери информации конечного порядка, если допускается обращение системы с запаздыванием. Автоматы Лоренца, как легко видеть, являются БПИ-автоматами в любом поле. Так как обычно речь идет об обработке информации с помощью компьютеров, то такая информация представляется последовательностью битов, более крупных единиц – байтов или блоков, кратных байтам по длине. В этом случае число различных элементов, описываемых всевозможными комбинациями значений отдельных битов, равно $2^m = q$, где $m = 8k$, $k \in N$. Поэтому соответствующие вычисления можно проводить либо в кольце Z_q, либо в поле $GF(2^m)$. Так как компьютерная обработка информации осуществляется побайтно, то реализацию автоматных аналогов удобно рассматривать в полях $GF(2^{8k})$, $k = 1, 2, ...$ В этом случаев в качестве базового рассматривается поле $GF(q) = GF(2^8)$ и неделимым элементом информации выступает байт. Такое поле строится как кольцо классов вычетов многочленов над полем $GF(2)$ по неприводимому над этим полем многочлену, например такому: $f(x) = x^8 + x^4 + x^3 + x^2 + 1$. Вычеты $A_0 + A_1x + A_2x^2 + A_3x^3 + A_4x^4 + A_5x^5 + A_6x^6 + A_7x^7$ по модулю соответствующего многочлена описываются как булевы векторы A_0, A_1, ..., A_7, где A_i равно 0 или 1, $i = 0, 1, ..., 7$. Эксперименты по шифрованию информации с помощью автоматов Лоренца позволили обнаружить следующий эффект. Если входное слово, подаваемое на автомат, подвергается искажениям (например, один из символов меняется на какой-то другой), то возможны два варианта: либо выходное слово, начиная с момента искажения, полностью изменяется, либо через некоторое число шагов после момента искажения оно совпадает с соответствующим конечным отрезком неискаженного выходного слова. Последнее свойство аналогично свойству самосинхронизируемости некоторых поточных шифрсистем [8] и говорит об определенной устойчивости автомата к искажениям входной информации. Этот эффект и наблюдался при расшифровывании искаженной информации автоматами Лоренца. В [6] на основе введенных понятий синхронизируемости и k-локальной синхронизируемости состояний описана структура некоторых автоматов, обладающих такими свойствами. Способность восстановления функционирования алгоритма преобразования после искажения входных последовательностей может либо поддерживаться (как, например, в самосинхронизирующихся поточных шифросистемах), либо подавляться, как свойство, ограничивающее распространение искажения одного символа на возможно большее число символов шифртекста. Такие особенности могут влиять на криптостойкость алгоритмов. В п. 2 описано явление деградации, связанное с вырождением множества траекторий динамической системы в результате попадания в некоторое подмножество состояний фазового пространства. В [6] показано, что в автоматном случае это свойство может быть следствием особенностей структуры графа переходов автомата, которую можно описать следующим образом. На множестве состояний автомата определяется специальная конгруэнция k-локальной синхронизируемости состояний, которая описывает свойство «устойчивости» к «искажениям» подслов фиксированной длины во входных последовательностях. Доказано, что факторизация по этой конгруэнции определяет фактор-автомат, в котором множество состояний распадается на непересекающиеся циклы. Уход с этих циклов возможен только при дополнительных управляющих воздействиях на автомат, переводящих его в новое состояние. Такое состояние, вообще говоря, может и не принадлежать исходному множеству состояний. Далее предлагается способ формирования таких воздействий, заключающийся в регулировании размерности пространства состояний и изменении, таким образом, исходного множества состояний.

4. Управление размерностью пространства состояний

Пусть $A = (X, U, Y, \delta, \lambda)$ – автомат Мили. Автомат $B = (X_B, U_B, Y_B, \delta_B, \lambda_B)$ будем называть подавтоматом автомата A (и писать $B \subseteq A$), если $X_B \subseteq X$, $U_B \subseteq U$, $Y_B \subseteq Y$, а δ_B и λ_B есть сужения соответственно функций δ и λ на множество $X_B \times U_B$.

Пусть автомат A описывается системой (11) над полем $GF(q)$. В этом случае автомат будем обозначать через A_q. Результаты теории конечных полей позволяют считать, что $GF(q)$ есть подполе поля $GF(q^n)$ при любом натуральном n. В силу этого уравнения (11) можно понимать как уравнения, задающие некоторый автомат A_q^n в поле $GF(q^n)$. Ограничения его функций на поле $GF(q)$, в силу замкнутости последнего, определяют подавтомат, изоморфный автомату A_q, который будем обозначать таким же образом. Пусть задана последовательность расширений поля $GF(q)$, $GF(q^{m_1})$, $GF(q^{m_2})$, …, $GF(q^{m_n})$, такая, что $m_i | m_{i+1}$, $i = 1, \ldots, n–1$. В этом случае $GF(q) \subset GF(q^{m_1}) \subset GF(q^{m_2}) \subset \ldots \subset GF(q^{m_n})$, что определяет, в силу вышесказанного, последовательность автоматов $A_q \subset A_q^{m_1} \subset \ldots \subset A_q^{m_n}$. Если же числа m_i, $i = 1, \ldots, n–1$, произвольны (в частности, попарно взаимно просты), то поле $GF(q^m)$, где m – наименьшее общее кратное этих чисел, содержит всякое подполе $GF(q^{m_i})$, а значит, имеется и вложение соответствующих автоматов. Более того, все сказанное справедливо и в случае, когда в качестве расширения поля $GF(q)$ выбирается его алгебраическое замыкание (обозначим его $GF(q^\infty)$). Тогда и соответствующие уравнения, задающие исходный автомат, можно понимать как уравнения в поле $GF(q^\infty)$, и эти уравнения задают уже бесконечный автомат A_q^∞. Из единственности алгебраического замыкания $GF(q^\infty)$ и вышесказанного следует

Утверждение. Для произвольных автомата A_q и натурального m справедливы включения $A_q \subset A_q^m \subset A_q^\infty$.

Сказанное обосновывает построение нового автомата $A(m_1, \ldots, m_n)$ фактически с переменными множествами состояний, входов и выходов. Его функционирование в любой момент времени совпадает с функционированием одного из автоматов $A_q^{m_i}$, $i = 1, \ldots, n$. Смена одного автомата другим или «принудительная» смена текущего состояния в процессе функционирования может осуществляться при выполнении некоторого предиката $P(x, p)$, определенного на $X \times U^*$, где X и U – множества состояний и входов соответственно текущего автомата A.

В качестве указанного предиката можно выбрать, например, условие появления деградации, описанное выше, условие попадания в определенные состояния или появление фиксированных подслов во входной последовательности. Например, для систем (8), (9) такой предикат можно определить как $P = ((x_1(t) = x_3(t)) \vee (x_1(t) \cdot x_2(t) \cdot x_3(t) = 0)) \wedge (u(t) \cdot u(t–1) \cdot u(t–2) = 0))$. Его истинностное значение определяет «принудительную» смену состояния, например, по такому правилу: $x_1(t+1) = u(t)$, $x_2(t+1) = u(t–1)$, $x_3(t+1)) = u(t–2)$, либо, если отказаться от условия неравенства нулю трех последовательных входных символов, по более простому правилу из п. 2: $x(t+1) = X(t+1) = x_0$.

Дополнительные вычисления для проверки истинности предиката, естественно, повышают сложность алгоритма преобразования информации. Это требует наложения ограничений на временную и емкостную сложности вычисления таких предикатов. В приведенном примере эту сложность легко оценить: при фиксации верхней границы размерности обе сложности оцениваются некоторыми константами. В целом это не повышает порядка оценок по сравнению с алгоритмом без вычисления предиката. В общем случае ситуация более сложна и требует исследования структуры автомата.

Пусть входная последовательность $p = u_1 u_2 \ldots u_k$ в некотором исходном алфавите $U = GF(q)$ подается на автомат $A(m_1, \ldots, m_n)$. Она обрабатывается этим автоматом либо посимвольно, либо блоками размера m_i, $i = 1, \ldots, n$, где размер блока определяется выбором поля, как указывалось выше. В результате обработки всей последовательности p она оказывается разбитой на подслова разной длины (блоки), каждое из которых преобразуется своим автоматом. Это разбиение заранее неизвестно и определяется предикатом $P(x, p)$. Если он существенно зависит от x, этот параметр или иной другой можно сделать секретным, в дополнение к секретным коэффициентам, задающим конкретный автомат-шифратор в каждом сеансе преобразования. В качестве такого предиката для автоматов Лоренца, учитывая возможность попадания на вышеуказанные циклы фактор-автомата, может быть выбрано, например, условие совпадения текущего состояния с заранее заданным состоянием, уже появлявшимся в один из предыдущих моментов времени, либо условие появления во входном слове заданного подслова. При выполнении этих условий происходит изменение размерности очередного обрабатываемого блока. Размер блока может выбираться из заранее оговоренного списка значений в порядке, который либо жестко фиксирован, либо снова может определяться некоторым предикатом. Если в списке все значения размерностей взаимно просты, то наименьшая размерность поля (как векторного пространства), содержащего все подполя выбранных размерностей, равна произведению этих размерностей. Это усложнит анализ поведения шифрующего автомата на основе выбора, в силу утверждения 1, в качестве исходного поля указанного надполя. Помимо этого, усложнение анализа поведения шифрующего автомата определяется также тем, что задача восстановления разбиения входного слова на заданные подслова, каждое из которых обрабатывается своим подавтоматом, относится к классу задач упаковки. Эта задача может решаться перебором, если заранее известен список возможных размерностей. Однако прямой перебор затруд-

нен, так как число вариантов указанных разбиений с ростом длины слова даже при небольших значениях m_i растет достаточно быстро.

Заключение

Предложенный метод динамического изменения размерности пространства состояний позволяет усложнять поведение системы без принципиального изменения ее описания и сложности алгоритма преобразования. Пополнение описания системы условиями, меняющими размерность пространства состояний, а не саму систему уравнений означает, что фактически таким образом вводятся, помимо параметрических, структурные ключи, которые также могут быть секретными. Это позволяет повысить стойкость соответствующих алгоритмов преобразования информации. Предложенный вариант обработки информации динамическими системами с переменной размерностью пространства состояний может быть полезным при разработке шифров с регулируемой степенью криптостойкости. При этом необходимы дополнительные исследования, связанные с вопросами формирования последовательностей примитивных полиномов для описания соответствующих полей и автоматов, с оценкой усложнения соответствующих алгоритмов и по памяти, и по времени, и др.

ЛИТЕРАТУРА

1. *Feldmann U.*, *Hasler M. and Schwarz W.* Communication by chaotic signals: the inverse system approach // Int. J. Circ. Theory Appl. 1996. V. 24. P. 551 – 579.
2. *Ковалев А.М.*, *Щербак В.Ф.* Управляемость, наблюдаемость, идентифицируемость динамических систем. Киев: Наукова думка, 1993. 285 с.
3. *Ковалев А.М.* Критерии функциональной управляемости и обратимости нелинейных систем // ПММ. 1998. Т. 62. Вып. 1. С. 110 – 120.
4. *Кудрявцев В.Б.*, *Алешин С.В.*, *Подколзин А.С.* Введение в теорию автоматов. М.: Наука, 1985. 320 с.
5. *Лидл Р.*, *Нидеррайтер Т.* Конечные поля. М.: Мир, 1988. Т. 1, 2. 820 с.
6. *Козловский В.А.*, *Толмачевская Л.А.* Автоматные аналоги динамических хаотических систем // Труды ИПММ НАН Украины. 2003. Т. 8. С. 59 – 69.
7. *Данилов Ю.А.* Лекции по нелинейной динамике. Элементарное введение. М.: Постмаркет, 2001. 184 с.
8. *Алферов А.П.*, *Зубов А.Ю.*, *Кузьмин А.С.*, *Черемушкин А.В.* Основы криптографии. М.: Гелиос АРВ, 2002. 480 с.