УДК 004.622

ИССЛЕДОВАНИЕ СИСТЕМЫ ПЕРЕДАЧИ ДАННЫХ С ШИФРОВАНИЕМ НА ОСНОВЕ ДИНАМИЧЕСКОГО ХАОСА

Сапожникова Ю.В., Андреев В.В.
Чувашский государственный университет им. И. Н. Ульянова, г. Чебоксары
Тел. +7 (902) – 287 – 05 – 12 E-mail: andreev_vsevolod@mail.ru

Аннотация. В работе разработан и исследован алгоритм кодирования данных на основе динамического детерминированного хаоса.
Ключевые слова: кодирование, помехоустойчивое кодирование, алгоритм, детерминированный хаос

Постановка задачи

В настоящее время во всем мире происходит бурное развитие вычислительных систем. Также растет количество информации, циркулирующей по этим системам. Все чаще человек обращается к помощи ЭВМ для хранения, преобразования и обработки своих конфиденциальных данных. В то же время растет количество злоумышленников, пытающихся получить доступ к этим данным. Для защиты данных от несанкционированного доступа было придумано множество методов, большинство из которых с ростом производительности процессоров становятся неустойчивыми к взлому грубой силой. Одним из подходов к решению этой проблемы является применение теории детерминированного хаоса [1, 2]. Детерминированный хаос представляет собой систему, поведение которой выглядит случайным, но определяется строго заданной системой уравнений. Поведение такой системы сильно зависит от начальных условий и других параметров.

Метод решения

В настоящей работе, как и в [3], в качестве генератора хаоса использован аттрактор Лоренца, описываемый системой дифференциальных уравнений [4]:

$$\begin{aligned} dX/dt &= -\sigma X + \sigma Y, \\ dY/dt &= -XZ + rX - Y, \\ dZ/dt &= XY - bZ. \end{aligned} \qquad (1)$$

Здесь $r = 28$; $\sigma = 10$; $b = 8/3$; t - время.

В работе разработаны и исследованы два алгоритма кодирования данных на основе динамического детерминированного хаоса.

Для шифрования сигнала использован следующий алгоритм.

1. Отсчёты исходного сигнала $s(t)$ сортируем по возрастанию. При этом, если среди отсчётов встречается несколько одинаковых, оставляем только один из них. В результате получим массив m1unique.
2. В массиве n_1 запоминаем номера отсчётов исходного сигнала, совпадающих по уровню с каждым из элементов массива m1unique (см. рис.1). Таким образом, размерность массива n_1 равна размерности массива $s(t)$.
3. В массиве n_2 запоминаем количество повторений в исходном сигнале каждого из отсчётов, записанных в массив m1unique (см. рис.1). Следовательно, массивы m1unique и n_2 имеют одинаковую размерность.
4. Умножаем каждый элемент массива m1unique на постоянный коэффициент k, т.е. вычисляем k*m1unique. В точках временной оси t, совпадающих с элементами массива k*m1unique, находим решения аттрактора Лоренца (1). При этом решения системы диф-

ференциальных уравнений (1) могут быть как положительными, так и отрицательными. Осуществляем сдвиг зависимости, например $X(t)$, в область положительных значений: $X(t)+\Delta$, где константа Δ выбирается из условия $X(t)+\Delta \geq 0$ для $\forall t$.

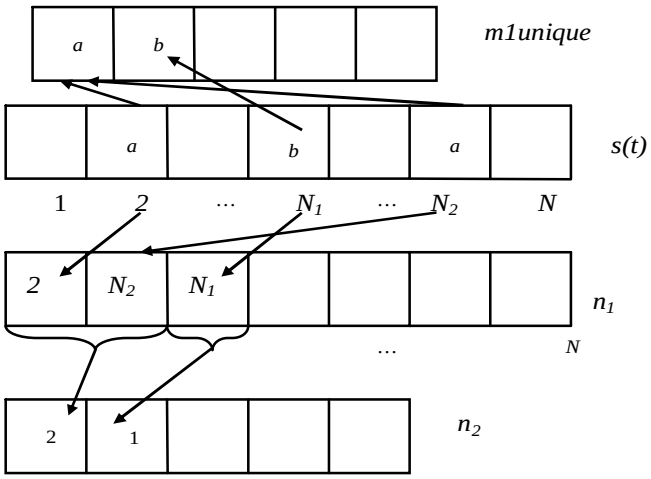

Рис. 1. К пояснению создания массивов m1unique, n_1, n_2

5. Нормируем отсчёты $X(t)+\Delta$ так, чтобы максимальный из них был равен размерности массива n_1. Так получим массив $X_1(t)$.
6. Округляем элементы массива $X_1(t)$.
7. Перебираем по порядку элементы массива $X_1(t)$. Допустим, очередной элемент с номером M_1 имеет значение a (см. рис.2). Тогда в массив n_1 после элемента с номером $a-1$ добавляем пустую ячейку, сдвигая при этом элементы, начиная с номера a, на одну единицу вправо. В добавленную ячейку записываем элемент с номером M_1 из массива n_2. В результате получим новый массив n_3 длиной $\dim(n_1+n_2)$.
8. В массив n_3 добавляем параметры аттрактора Лоренца и значение коэффициента k.

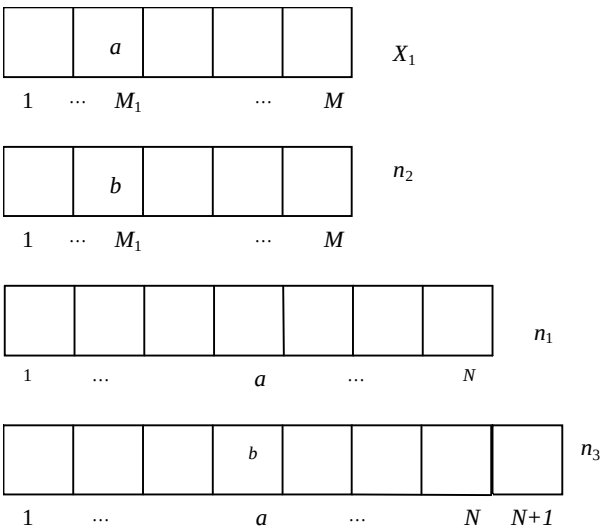

Рис. 2. К пояснению создания массива n_3

Декодирование сигнала в приёмнике происходит в следующем порядке.

1. На входе приёмника имеем коэффициент k, параметры аттрактора Лоренца (1), а также массивы m1unique и n_3.

2. Запускаем аттрактор Лоренца (1) и получаем его решение $X(t)$ в моменты времени, соответствующие элементам массива k *m1unique.
3. Производим нормирование элементов массива $X(t)$ в том же порядке, что и при шифровании. В результате получим массив $X_1(t)$.
4. Округляем элементы массива $X_1(t)$ так же, как и при шифровании.
5. Зная отсчёты $X_1(t)$, выделяем из массива n_3 массивы n_1 и n_2.
6. Зная массивы n_1, n_2 и m1unique восстанавливаем исходный сигнал $s(t)$.

Для данного алгоритма размер зашифрованного сигнала становится в 1.5 раза больше по сравнению с исходным, что является достаточно существенным. В то же время, когда речь идёт об обеспечении высокой степени устойчивости метода шифрования по отношению к несанкционированным попыткам взлома, такое увеличение объёма передаваемых данных может быть вполне терпимым. Здесь взломщик, не обладая генератором детерминированного хаоса, использованным при кодировании данных, практически не сможет правильно их декодировать методом перебора.

Для уменьшения объема зашифрованного сигнала вместо массива m1unique через канал связи можно передавать коэффициенты аппроксимирующего его полинома.

Результаты

Для исследования устойчивости алгоритма к различным помехам был смоделирован канал связи (рис. 3) в среде Matlab.

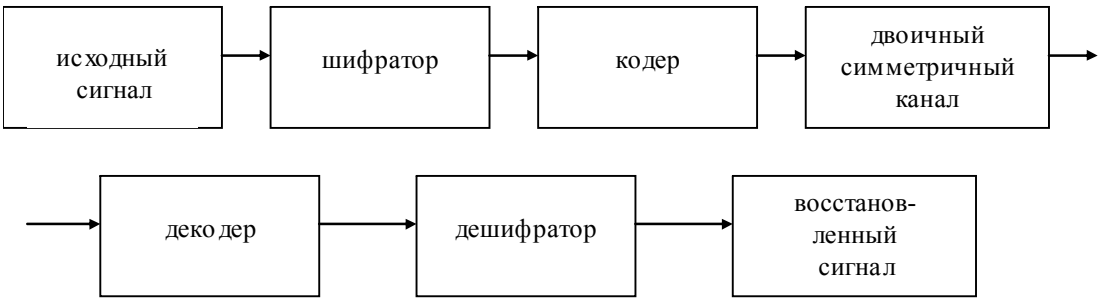

Рис. 3. Блок- схема канала связи

Исходный сигнал поступает на вход шифратора, который преобразует сигнал в соответствии с представленным выше алгоритмом. После этого данные поступают на вход следующего блока, реализующего помехоустойчивое кодирование по методу Хэмминга. Кодированные сообщения пропускаются через двоичный симметричный канал с вероятностью битовой ошибки p. В приёмнике после прохождения декодера и дешифратора получаем восстановленный сигнал, который при отсутствии ошибок в канале передачи данных должен быть идентичен исходному.

В работе было исследовано соответствие исходным восстановленных коэффициентов полинома, аппроксимирующих массив m1unique, после прохождения зашифрованным сигналом двоичного симметричного канала связи. Исследование проводилось для различных вероятностей битовой ошибки и количества проверочных символов в коде Хэмминга (см. табл.1). Исходные коэффициенты полинома равны 0; 0; 0; 0; 0; 0; 0; 0; 0,0022; -0,2414. Следовательно, порядок исходного полинома равен 9.

С увеличением количества проверочных символов в коде Хэмминга происходит увеличение длины передаваемых сообщений. Следовательно, вероятность возникновения ошибок за время передачи данных также возрастает. Это видно из результатов, представленных в табл. 1.

Таким образом, при вероятности ошибки, меньшей 0.01, и применении помехоустой-

чивого кодирования Хэмминга исходный сигнал восстанавливается точно. А уже при вероятности битовой ошибки, большей 0.05 исходный сигнал точно не восстанавливается даже при увеличении количества проверочных символов в коде Хэмминга. К тому же при этом существенно увеличивается время обработки.

Таблица 1.

№ п/п	Вероятность ошибки p	Количество проверочных символов m	Время вычислений t в секундах	Коэффициенты полинома на выходе приёмника
1	0.001	3	0.0150	0; 0; 0; 0; 0; 0; 0; 0; 0,0022; -0,2414
2	0.01	3	0.0160	0; 0; 0; 0; 0; 0; 0; 0; 0,0022; -0,2414
3	0.05	3	0.0160	0.1536; 0; 0; 0; 0; 0; 0; 0; 0.0022; -0.2414
4	0.1	3	0.0150	2.1856; 0; 0; 0.0539; -2.0479; 0; 0; 0.0278; -0.2414
5	0.001	4	0.0160	0; 0; 0; 0; 0; 0; 0; 0; 0,0022; -0,2414
6	0.01	4	0.0160	0; 0; 0; 0; 0; 0; 0; 0; 0,0022; -0,2414
7	0.05	4	0.0160	157.4912; 0; 124.526; 0.128; 105.6768; 0; 0; 0; 0.0022; -0.3407
8	0.1	4	0.0160	7.7824; 0.0387; 9.0E-4; 0; -209.7151; 0; 0.0532; 0; -180.8609; 153.9731
9	0.05	5	0.0310	0; 0; 1678.0296; 0; 0; -3348.8893; 0; 81.9712; 72.0978; -0.2414
10	0.05	7	0.0620	-6.64613997892458E31; -4.984604957875119E31; 2.0302216651658936E25; 4.8028785555547303E30; 6.541077097204596E28; 1.5845662725998364E25; 0; -6.646139780854171E31; 6.338301358185461E25; -1.03845937367147E30
11	0.1	5	0.0320	0; 0; 1678.0296; 0; 0; -3348.8893; 0; 81.9712; 72.0978; -0.2414
12	0.1	7	0.0470	7.897519343968195E15; 3.093077680282439E29; 2.0698444575068443E27; 8.112979315947422E27; 8.51455555161303E30; 1.1033670999598246E30; 2.1867437100783876E27; 1.2701313160846627E26; 5.192296897371609E29; -1.7240295872240492E28

Литература

1. Lepsoy S., Oien G.E., Ramstad T.A. Attractor image compression with a fast non-iterative decoding algorithm. In book: Acoustics, Speech, and Signal Processing: 1993 IEEE International Conference. 1993. V.5. P. 337−340.
2. Kal'yanov G.N., Kal'yanov E.V. Coding Digital Information with the Use of Generators with Chaotic Dynamics // Journal of communications technologies and electronics. 2008. №4. P. 459−467.
3. Сапожникова Ю.В., Андреев В.В. Хаос и кодирование информации. В кн.: Информационные технологии в профессиональной деятельности и научной работе: сборник материалов Всероссийской научно-практической конференции с международным участием. Йошкар-Ола: Марийский государственный технический университет: в 2 ч. – Ч.2. 2009. 184 с. С. 9−12.
4. Шустер Г. Детерминированный хаос: введение. – М.: Мир, 1988. – С. 240.