$$1 - A - 3\sqrt{\frac{A(1-A)}{m}} \le \frac{k}{m} \le 1 - A + 3\sqrt{\frac{A(1-A)}{m}}\ . \qquad (3)$$

Якщо виконуються нерівності (1), (2), (3), то генератор будемо вважати генератором суто випадкових послідовностей.

Якщо такий генератор використовується у криптографічних модулях, до яких висуваються вимоги 3 – 4 рівнів безпеки [10], то NIST рекомендує також проводити тестування генераторів при запуску (рівень 4) та при необхідності в будь-який інший момент (рівні 3, 4). Тестування полягає у застосуванні до перших 20 000 бітів згенерованої послідовності наступних тестів (при α=0,001):

- монобітний тест (тобто критерій χ^2);
- тест покеру (критерій χ^2 для 4-грам, що не перекриваються);
- тест серій (критерій кількості *S*-серій);
- тест серій максимальної довжини.

Крім того, протягом роботи генератора постійно має виконуватись наступний тест. Починаючи з 1-го біту, згенерована послідовність розбивається на *n*-грами ($n \ge 16$). Кожна наступна *n*-грама не повинна дорівнювати попередній.

Література: 1. T. M. Cover, G. A. Thomas, Elements of information theory, Wiley, New-York, 1991. 2. M. Li, P. Vitanyi, An introduction to Kolmogorov complexity and its applications, Springer-Verlag, New-York, 1993. 3. M. Blum, S. Micali, How to generate cryptographically strong sequences of pseudo-random bits, SIAM J. Comput. 13(1984), p. 850–864. 4. A. C. Yao, Theory and applications of trap-door functions, 23rd IEEE Symposium on Foundations of Computer Science, 1982, pp. 80–91. 5. Л. В. Ковальчук, О периодах выходных последовательностей некоторых псевдослучайных генераторов, построенных на основе односторонних функций, III Международная научно-практическая конференция "Безопасность информации в информационно - телекоммуникационных системах", Киев, 2000. 6. O. Goldreich, Pseudorandomness, Notices of AMS, v. 46, № 10, 1999, p. 1209–1216. 7. О. Вербицький, Вступ до криптології, Видавництво науково-технічної літератури, Львів, 1998, 247 с. 8. NIST Special Publications 800–22, A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications, 2000. 9. U. M. Maurer, A Universal Statistical Test for Random Bit Generators, Journal of Cryptolojy, № 5, 1992, p. 89–105. 10. Federal Information Processing Standard Publications "Sequrity Requirements for Cryptographic Modules" 140–1, January 11, 1994, 50 p. 11. Ю. С. Харин, В. И. Берник, Г. В. Матвеев, Математические основы криптологии, Минск, БГУ, 1999, 319 с.

УДК 681.3.06

ПОСТРОЕНИЕ ЛИНЕЙНЫХ БАЗИСОВ ВЕКТОРНЫХ ПРОСТРАНСТВ НАД ПОЛЕМ GF(2)

Сергей Головашич

Харьковский национальный университет радиоэлектроники

Анотація: **Пропонуються два алгоритми побудови лінійних базисів над простором $GF(2)^n$. Перший алгоритм дозволяє здійснювати вибір з повної множини базисів над $GF(2)^n$, але є відносно складним. Другий є значно простішим, але здійснює вибір не з повної множини.**

Summary: **Two algorithms for building a linear basis over $GF(2)^n$ are proposed. The first one allows choosing a linear basis of the full space, but it is comparatively complicated. The second one is much easier, but it performs choosing from the truncated space.**

Ключевые слова: **Криптография, линейное отображение, базис линейного пространства, невырожденная (не сингулярная) матрица.**

Процедуру блочного симметричного шифрования (БСШ) можно рассматривать как композицию множества элементарных отображений всего блока данных либо его фрагментов. Эти элементарные отображения можно разделить на три группы:

- фиксированные – отображения, постоянные для всех блоков данных и ключей шифрования (например, структурные схемы SPN или SLTN-преобразований, нелинейные преобразования – S-блоки);

- статически управляемые – отображения, которые формируются предварительно, на основе долговременного ключа шифрования, если таковой используется (например, сменные секретные S-блоки);
- динамически управляемые – отображения, которые формируются в реальном масштабе времени на основе сеансового ключа и/или промежуточных данных (например, сложение или умножение на подключ по некоторому модулю, управляемые циклические сдвиги и т. д.).

Объединение данных с секретным ключом в БСШ выполняется посредством элементарных управляемых отображений, при этом вычислительная сложность шифра существенно зависит от сложности этих преобразований. Поэтому при выборе управляемых отображений разработчики обычно стремятся достигнуть компромисса между вносимой неопределённостью, необходимым объёмом ключевой (управляющей) информации и сложностью реализации преобразования.

Информацию, используемую для выбора отображения из некоторого предопределённого множества, сопоставленного с *управляемым отображением*, будем называть *управляющим сигналом*. В качестве источника этой информации могут выступать как секретный ключ, так и промежуточные данные.

Наибольшее распространение в силу простоты реализации на широком спектре аппаратных средств получили простые аффинные операции: сложение подблока фиксированной длины n с подключом той же длины, по модулю равному 2 либо 2^n. Эти операции можно рассматривать как управляемые отображения с размерностью поддерживаемого множества 2^n. С другой стороны, отмеченные бинарные операции на множестве допустимых значений образуют группу. Если полноцикловая процедура шифрования с вероятностью, большей чем вероятность «угадывания» ключа аппроксимируется групповой операцией, используемой для введения ключа, то этот криптоалгоритм становится уязвим к атакам дифференциального криптоанализа [1]. Основным методом защиты от криптоатак этого типа является комбинирование различных групповых операций и применение высоко нелинейных преобразований (S-блоков), которые могут быть как открытыми, так и секретными [2] (в последнем случае сложность криптоанализа увеличивается). В качестве альтернативного либо дополнительного средства защиты некоторые разработчики предлагают использовать управляемый циклический сдвиг подблоков [3, 4].

Отметим, что циклический сдвиг является частным случаем перестановки битов, и, следовательно, частным случаем обратимого линейного преобразования над $GF(2)^n$. Основным достоинством управляемых линейных отображений перед другими управляемыми преобразованиями над $GF(2)^n$ является простота как аппаратной, так и программной реализации. Так произвольное линейное отображение $\mathrm{F}_2^n \to \mathrm{F}_2^m$ может быть задано в виде матричного умножения:

$$\mathrm{Y} = \mathrm{A} \times \mathrm{X},$$

где X – n-мерный входной вектор-столбец;
Y – m-мерный выходной вектор-столбец;
A – матрица размерности $(m \times n)$, $\mathbf{A} = \{\alpha_{i,j}\}$, $\alpha_{i,j} \in \mathbf{F}_2$, $i = \overline{1,m}$, $j = \overline{1,n}$.

Такое преобразование (умножение матрицы на вектор) может быть эффективно реализовано программно и особенно аппаратно, т. к. сводится к последовательному выполнению n пар простейших логических операций AND и XOR над m-мерными векторами, каждый из которых получается m-кратным тиражированием значения соответствующего разряда входного вектора X, при этом вычисление всех разрядов вектора результата может выполняться одновременно:

$$y_i = \bigoplus_{j=1}^{n} \alpha_{i,j} \wedge x_j, \quad i = \overline{1,m}, \ \mathrm{X} = (x_1,...,x_n) \in \mathrm{F}_2^n, \ \mathrm{Y} = (y_1,...,y_m) \in \mathrm{F}_2^m$$

$$\mathrm{Y} = \bigoplus_{j=1}^{n} \mathrm{T}_j \wedge \mathrm{Z}_j, \quad \mathrm{T}_j = (\alpha_{1,j},...,\alpha_{m,j}) \in \mathrm{F}_2^m, \ \mathrm{Z}_j = (x_j,...,x_j) \in \mathrm{F}_2^m$$

Кроме того, при программной реализации несколько матричных умножений (для малых m) может быть распараллелено.

Обратимые (биективные) линейные преобразования над F_2^n соответствуют умножению на квадратные ($m = n$) невырожденные (не сингулярные) матрицы, т. е. квадратные матрицы, для которых определитель отличен от 0. Строки (и столбцы) таких матриц линейно независимы, и образуют базис над F_2^n, а само линейное преобразование эквивалентно замене базиса векторного пространства.

Очевидно, что применение обратимых аффинных преобразований общего вида позволяет внести большую неопределённость об управляющем сигнале (ключе) по сравнению с простыми аффинными преобразованиями (сложением либо циклическим сдвигом). При этом вносимая управляемым преобразованием неопределённость будет прямо пропорциональна размерности поддерживаемого множества отображений и достигнет максимума для заданной разрядности n, когда это множество будет содержать все элементы полной линейной группы $GL(n, F_2)$.

При построении обратимых управляемых линейных отображений определённую сложность представляет задача выбора невырожденных матриц из полного множества квадратных матриц порядка n, что эквивалентно поиску всех различных линейных базисов (с точностью до порядка расположения векторов) на заданном векторном пространстве. Для многих приложений вероятностные методы формирования таких матриц, т. е. основанные на «фильтрации» случайно (псевдослучайно) сформированных квадратных матриц, являются неприемлемыми, т. к. время ожидания «успешного» результата для этих методов становится случайной величиной. В связи с этим особый интерес представляют детерминированные алгоритмы выбора «по индексу» некоторой матрицы из полного множества невырожденных матриц заданного порядка (либо из некоторого его подмножества близкой размерности).

В работе рассматривается два алгоритма построения линейных базисов над векторными пространствами $GF(2)^n$, адаптированных соответственно для приложений статического и динамического формирования линейных преобразований.

I Выбор линейного базиса из полного пространства

Прежде всего заметим, что общее количество матриц размерности $n \times n$ с элементами из GF(2) равно $N = 2^{n^2}$, из них невырожденных матриц

$$N_{nsm} = \prod_{i=0}^{n-1}\left(2^n - 2^i\right),$$

что следует из условия о том, что каждая последующая строка (столбец) квадратной матрицы не является линейной комбинацией предыдущих (и не равна 0). Из этого соотношения можно получить оценку количества битов, необходимых для определения всех невырожденных матриц порядка n с элементами F_2

$$\lceil \log_2 N_{nsm} \rceil = n^2 - 1,$$

и сделать заключение, что $N/4 < N_{nsm} < N/2$, где N – общее количество квадратных матриц требуемого порядка n.

Для обозначения различных типов матриц воспользуемся следующими обозначениями [5]: I – единичная матрица, $\mathrm{L} = \{l_{i,j}\}$ – нижняя треугольная матрица, $\mathrm{U} = \{u_{i,j}\}$ – верхняя треугольная матрица. Воспользуемся также понятием матрицы *перестановки* P, под которой будем понимать квадратную матрицу, содержащую только по одному ненулевому элементу (единице) в каждой строке и столбце. При умножении некоторой матрицы A на матрицу P справа ($\mathrm{A} \times \mathrm{P}$) равносильно перестановке столбцов матрицы A, а при умножении слева ($\mathrm{P} \times \mathrm{A}$) – перестановке строк A.

Напомним, что любая треугольная матрица является невырожденной, т. к. очевидно, что все её строки (столбцы) линейно независимы, а определитель равен произведению элементов главной диагонали, т. е. для случая GF(2) равен 1:

$$\det(\mathrm{L}) = \prod_{i=0}^{n-1} l_{i,i}, \quad \det(\mathrm{U}) = \prod_{i=0}^{n-1} u_{i,i}, \quad l_{i,i} = u_{i,i} = 1, \quad i = \overline{0, n-1}.$$

Более того, любая матрица перестановки P также является невырожденной, т. к. может быть преобразована к единичной матрице I путём некоторой перестановки строк (столбцов).

Искомый алгоритм выбора невырожденных матриц из полного множества квадратных матриц порядка n может быть построен на основе следующего свойства [5].

Лемма 1. Любая квадратная невырожденная матрица A может быть представлена матричным произведением $\mathrm{A} = \mathrm{P} \times \mathrm{L} \times \mathrm{U}$, где P – матрица перестановки, L – нижняя треугольная матрица, U – верхняя треугольная матрица.

Напомним, что произведение невырожденных матриц некоторого порядка также равно невырожденной матрице того же порядка, т. к.:

$$\det(\mathrm{A} \times \mathrm{B}) = \det(\mathrm{A}) \times \det(\mathrm{B}).$$

Из свойств матричного произведения вытекает следующая лемма.

Лемма 2. Произведение двух треугольных матриц одного порядка и одного типа (верхних либо нижних) также является треугольной матрицей того же типа, а произведение двух треугольных матриц различного типа, не равных единичной, является не треугольной матрицей.

Следствие. Матрица, обратная треугольной матрице, является треугольной матрицей того же типа.

На основе предыдущей леммы можно доказать следующую.

Лемма 3. Матричные произведения $\mathrm{L} \times \mathrm{U}$ все различны для различных сомножителей, т. е. $\mathrm{L}_1 \times \mathrm{U}_1 \neq \mathrm{L}_2 \times \mathrm{U}_2$, если $\mathrm{L}_1 \neq \mathrm{L}_2$ и/или $\mathrm{U}_1 \neq \mathrm{U}_2$, где L_1, L_2 – нижние треугольные матрицы, U_1, U_2 – верхние треугольные матрицы.

Доказательство. Допустим, что $\mathrm{L}_1 \times \mathrm{U}_1 = \mathrm{L}_2 \times \mathrm{U}_2$. Определим две матрицы L_3 и U_3 такие, что

$$\mathrm{L}_2 = \mathrm{L}_1 \times \mathrm{L}_3 \text{ и } \mathrm{U}_2 = \mathrm{U}_3 \times \mathrm{U}_1,$$

$$\text{т. е. } \mathrm{L}_3 = \mathrm{L}_1^{-1} \times \mathrm{L}_2 \text{ и } \mathrm{U}_3 = \mathrm{U}_2 \times \mathrm{U}_1^{-1},$$

при этом в силу леммы 2 получаем, что L_3 и U_3 являются соответственно нижней и верхней треугольными матрицами. Подставляя выражения для L_2 и U_2 в исходное соотношение получим:

$$\mathrm{L}_1 \times \mathrm{U}_1 = \mathrm{L}_1 \times \mathrm{L}_3 \times \mathrm{U}_3 \times \mathrm{U}_1$$

откуда следует, что

$$\mathrm{L}_3 \times \mathrm{U}_3 = \mathrm{I},$$

а так как равенство $\mathrm{L}_3 = \mathrm{U}_3^{-1}$ невозможно для $\mathrm{L}_3 \neq \mathrm{U}_3 \neq \mathrm{I}$ в силу леммы 2, получаем $\mathrm{L}_3 = \mathrm{U}_3 = \mathrm{I}$, откуда $\mathrm{L}_2 = \mathrm{L}_1$ и $\mathrm{U}_2 = \mathrm{U}_1$, что противоречит исходному положению $\mathrm{L}_1 \neq \mathrm{L}_2$ и/или $\mathrm{U}_1 \neq \mathrm{U}_2$.

На основе последней леммы можно определить следующий простой алгоритм формирования 2^{n^2-n} различных невырожденных матриц порядка n:

1) «управляющая последовательность» $\mathrm{C} = \{c_k\}$ длиной $n^2 - n$ рассматривается как две треугольные матрицы: нижняя $\mathrm{L} = \{l_{i,j}\}$ и верхняя $\mathrm{U} = \{u_{i,j}\}$:

$$l_{i,j} = \begin{cases} 0, & для\ i < j \\ 1, & для\ i = j \\ c_{i\times(n-1)+j}, & для\ i > j \end{cases}, \quad u_{i,j} = \begin{cases} c_{i\times(n-1)+j}, & для\ i < j \\ 1, & для\ i = j \\ 0, & для\ i > j \end{cases}, \quad i,j = \overline{0,n-1}$$

2) выполнить умножение полученных матриц: $\mathrm{B} = \mathrm{L} \times \mathrm{U}$.

Достоинствами данного алгоритма является сравнительная простота реализации, однозначное соответствие «управляющего сигнала» результирующей матрице, а также то, что количество решений является степенью 2, т. е. двоичная «управляющая последовательность» C является безизбыточной.

Однако, множество невырожденных матриц, поддерживаемое приведенным алгоритмом, является неполным. Поэтому, если требуется выбор линейного преобразования A из полного множества, то в соответствии с леммой 1 все оставшиеся невырожденные матрицы можно получить перестановкой строк матрицы B, полученной указанным выше способом, т. е. $\mathrm{A} = \mathrm{P} \times \mathrm{B}$, где P – матрица перестановки, заданная дополнительной «управляющей последовательностью». Такой метод выбора линейных преобразований из полного множества является простым в реализации, однако обладает избыточностью в силу сложности выделения из полного множества $n!$ матриц перестановки порядка n только необходимого подмножества.

В тех случаях, когда совместно с прямым линейным преобразованием A, полученным указанным выше методом, также применяется и обратное к нему A^{-1}, для вычисления обратной матрицы может использоваться соотношение:

$$\mathrm{A}^{-1} = \mathrm{U}^{-1} \times \mathrm{L}^{-1} \times \mathrm{P}^{\mathrm{T}},$$

основанное на следующих свойствах матриц:

$$(\mathrm{A} \times \mathrm{B})^{-1} = \mathrm{B}^{-1} \times \mathrm{A}^{-1} \text{ и } \mathrm{P}^{-1} = \mathrm{P}^{\mathrm{T}},$$

где P^{T} – транспонированная матрица перестановки P (с элементами из GF(2)). Отметим, что задача обращения треугольных матриц (L и U) имеет более чем в 2 раза меньшую вычислительную сложность по сравнению с общим случаем. Поэтому выигрыш в производительности от применения указанного равенства будет существенен только в тех случаях, когда матрица A не хранится, и на момент вычисления A^{-1} известна только «управляющая последовательность» C.

Рассмотренный метод выбора линейных базисов использует вычислительно ёмкую операцию умножения треугольных матриц, что может оказаться неприемлемым для ряда приложений. Кроме того, алгоритм предполагает наличие «управляющего сигнала» относительно большой длины. Более того, значения указанных параметров возрастают, если выбор осуществляется из полного множества базисов. Поэтому, основной областью применения этого алгоритма является статическая генерация линейных отображений.

II Выбор линейного базиса из усеченного пространства

В ряде приложений к линейным преобразованиям могут предъявляться повышенные требования по производительности процедуры формирования линейного базиса при одновременном ограничении длины «управляющей последовательности». В таких случаях для формирования линейных базисов может использоваться аппарат теории конечных полей.

Если интерпретировать входной и выходной n разрядные векторы X и Y как полиномы над полем F_2 степени меньше n, принадлежащие факторкольцу $F_2[x]/(f)$, где $f(x)$ – произвольный (ненулевой) многочлен степени n из $F_2[x]$, то линейное биективное отображение $F_2^n \to F_2^n$ может быть задано следующим образом [7]:

$$h(x)\ \alpha\ \ g(x)\times h(x) mod\ f(x)$$

где $h(x)$ – многочлен степени меньше n, с коэффициентами, заданными входным вектором X, а $g(x)$ – фиксированный многочлен степени меньше n, взаимно-простой с многочленом $f(x)$.

Для указанного отображения множество частичных функций выхода $y_i = w_i(X)$ и входа $x_i = v_i(Y)$, где $i = 0, ..., n-1$, $X = (x_0, ..., x_{n-1})$, $Y = (y_0, ..., y_{n-1})$ являются линейными и образуют базисы на F_2^n.

В частности, если многочлен $f(x)$ является неприводимым, то факторкольцо $F_2[x]/(f)$ является полем [7], и в этом случае указанное отображение будет линейным автоморфизмом для произвольного, отличного от нуля полинома $g(x)$ степени меньше n. Поэтому, для неприводимого $f(x)$ и произвольного $g(x) \neq 0$, если мы имеем некоторый линейный базис на F_2^n: $(\alpha_0, ..., \alpha_{n-1})$, то последовательность векторов $(\beta_0, ..., \beta_{n-1})$, полученная как

$$\beta_i(x) = g(x)\times \alpha_i(x) mod\ f(x), \quad i = \overline{0,n-1}$$

также будет образовывать линейный базис над F_2^n.

Применяя в качестве исходного базиса $(x^0, x^1, ..., x^{n-1})$, получим

$$\beta_i(x) = g(x)\times x^i mod\ f(x), \quad i = \overline{0,n-1},$$

откуда вытекает следующий простой алгоритм формирования линейных базисов:

1) на основе «управляющего сигнала» сформировать отличный от 0 вектор v_g длиной n, а также выбрать из таблицы неприводимых полиномов $f(x)$ степени n, вектор v_f, соответствующий коэффициентам одного из этих полиномов;
2) принять $v_0 = v_g$;
3) вычислить очередной вектор базиса: $v_i = v_{i-1} << 1$ (где $[v << 1]$ – сдвиг двоичного вектора v на один разряд влево) и $v_i = v_i \oplus v_f$, если n-й разряд вектора β_i равен 1 (нумерация с 0 влево);
4) повторить пункт (3) для $i = 1, ..., n-1$.

Полученная в результате выполнения этого алгоритма совокупность векторов $(v_0, ..., v_{n-1})$ будет образовывать линейный базис над F_2^n.

Если для линейного отображения, полученного таким образом, необходимо определить обратное отображение, то следует вычислить обратный элемент $g^{-1}(x)$ в поле по модулю неприводимого многочлена $f(x)$ и применить его вместо $g(x)$ в соответствии с описанным выше алгоритмом. Для малых значений n вычисление обратного элемента по различным модулям $f(x)$ может быть реализовано таблично. Так, в частности, для $n = 8$, объём таблицы для одного модуля составит 256 байт, а учитывая, что количество различных неприводимых полиномов 8-й степени составляет 30, получим, что общий объём таблицы для всех модулей составит 7,5 KB. В приложениях, где требуется построение базиса для обратного отображения и накладываются ограничения на объём таблицы (и производительность), в качестве модуля может использоваться только один неприводимый полином. Кроме того, при аппаратной реализации вычисления в поле F_{2^n} могут быть реализованы особенно эффективно.

Следует отметить, что совокупность векторов линейного базиса, полученного рассмотренным способом, образует n линейно независимых перестановок относительно *генератора* v_g (при условии допустимости значения $v_g = 0$), что является необходимым условием для конструирования криптографически привлекательных преобразований – класса бент-отображений Маиораны-Макфарленда [8, 9].

С целью сокращения аппаратных затрат (использования имеющихся ресурсов) для формирования базиса может использоваться похожий алгоритм на основе n разрядного линейного рекуррентного регистра (ЛРР) с примитивным полиномом обратной связи [10]. В этом случае на основе «управляющего сигнала» может выполняться не только начальное заполнение регистра (отличное от 0), но и выбор характеристического многочлена из предопределённой таблицы примитивных полиномов степени n. Для формирования линейного базиса выполняется n–1 шаг преобразования состояния выбранного ЛРР. Полученные n соседних фаз регистра являются линейно независимыми и соответственно образуют базис над F_2^n, т. к. любые $2n$ соседних элементов линейной рекуррентной последовательности максимального периода образуют невырожденную систему линейных уравнений порядка n.

Алгоритм на основе ЛРР имеет сложность, эквивалентную рассмотренному выше алгоритму (n–1 пара операций – сдвиг и сложение по модулю 2), однако в отличие от предыдущего алгоритма способен формировать базисы из меньшего пространства, т. к. количество примитивных полиномов меньше количества неприводимых полиномов той же степени (например, для случая $n = 8$, имеем 16 примитивных против 30 неприводимых). Кроме того, для алгоритма на базе ЛРР процедура формирования обратного отображения является более сложной.

Заключение

Два рассмотренных выше алгоритма выбора линейных базисов предназначены для решения задачи формирования обратимых линейных преобразований над пространством F_2^n. Первый алгоритм основан на процедуре разложения квадратных матриц в произведение двух треугольных матриц и матрицы перестановки. Этот алгоритм позволяет каждому значению двоичного «управляющего сигнала» длиной n^2–n поставить в соответствие уникальный линейный базис. При этом если требуется выбор из полного множества, то алгоритм следует дополнить управляемой перестановкой векторов полученного базиса. В этом случае применение независимых «управляющих сигналов» для формирования базовой матрицы B и выбора перестановки её строк (из полного множества $n!$) является избыточным решением, что является платой за простоту реализации. Т. к. этот алгоритм содержит умножение матриц и управляемые перестановки строк (для выбора из полного множества), то его производительность неприемлема для многократного применения в составе цикловых функций современных БСШ. Основной областью применения этого алгоритма является статическое (предварительное) формирование секретных преобразований.

Второй алгоритм основан на мультипликативных свойствах элементов конечного поля GF(2^n) и отличается от первого предельной простотой реализации. Так для формирования линейного базиса на GF(2^n) требуется n–1 пара операций сдвига на 1 разряд и управляемого сложения по модулю 2 (XOR) n-разрядных слов. Этот алгоритм, по сравнению с предыдущим, использует гораздо меньше управляющей информации, поэтому выбор осуществляется из значительно ограниченного множества базисов. Второй алгоритм ориентирован на применение в схемах динамического формирования базисов (в реальном масштабе времени). Подобный алгоритм может использоваться в составе цикловой функции для определения управляемых линейных преобразований.

Литература: 1. E. Biham and A. Shamir. Differential Cryptanalysis of DES-like Cryptosystems. Journal of Cryptology, Vol. 4, No. 1, pp. 3–72, 1991. 2. ГОСТ 28147-89. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования. М.: Госстандарт СССР. 3. R. L. Rivest, «The RC5 encryption algorithm», B. Prenel editor, Fast Software Encryption, Second International Workshop (LNCS 1008), pp. 86–96, Springer-Verlag, 1995. 4. J. Borst. «The Block Cipher: Grand Cru». Available at http://www.cryptonessie.org. 5. Сигорский В. П. Математический аппарат инженера. Изд. 2-е стереотип. Изд-во «Техніка», 1977, 768 с. 6. Математическая энциклопедия: Гл. ред. И. М. Виноградов, М.: «Советская Энциклопедия», 1982. 7. Лидл Р., Нидеррайтер Г. Конечные поля: В 2-х т. Пер. с англ. – М.: Мир, 1988. – 822 с. 8. R. L. McFarland, "A family of difference sets in non-cyclic groups," J. Combinatorial Theory, Ser. A, 15, pp. 1–10, 1973. 9. K. Nyberg, "Construction of bent functions and difference sets," in Lecture Notes in Computer Science 473; Advances in Cryptology: Proc. Eurocrypt'90, I. Damgard, Ed., Aarhus, Denmark, May 21–24. 1990, pp. 151–160. Berlin: Springer-Verlag. 10. K. Nyberg, "New Bent Mappings Suitable for Fast Implementation", In R.

Anderson, editor, Fast Software Encryption – Cambridge Security Workshop, pp. 179–184, Springer-Verlag, Berlin, 1994.

УДК 621.391, 519.7

АЛГЕБРАИЧЕСКИЕ МОДЕЛИ КРИПТОГРАФИЧЕСКИХ СИСТЕМ С ОТКРЫТЫМ КЛЮЧОМ

Антон Алексейчук, Сергей Пришлин, Александр Романов
Военный институт телекоммуникации и информатизации НТУУ "КПИ"

Анотація: **Запропоновані загальні алгебраїчні моделі систем відкритого шифрування інформації та систем цифрового підпису, які є багатоосновними універсальними алгебрами. В межах розроблених моделей стисло охарактерізовані основні задачі аналізу стійкості систем з відкритим ключем, розглянуті взаємозв'язки між системами цифрового підпису і відкритого шифрування, а також приклади застосування алгебраїчних моделей цих систем.**
Summary: **There were proposed the general algebraic models of the open encryption systems and digital signature that are the multybase universal algebras. Within a framework of the designed models the general problems of the analysis of the security of the public-key systems were briefly characterized, the connections between the systems of public-key encryption and digital signature were considered. The examples of practical using of these algebraic models were given.**
Ключові слова: **Система відкритого шифрування, система цифрового підпису, багатоосновна алгебра.**

I Введение

После выхода в свет в 1976 году известной работы У. Диффи и М. Хеллмана [1], в которой впервые были изложены принципы построения систем открытого шифрования и открытого распределения ключей, появилось большое число различных вариантов таких систем, многие из которых послужили источниками новых математических задач. В 1994 году В. А. Артамонов и В. В. Ященко [2] предложили единую алгебраическую модель процедуры выработки общего ключа (так называемую public key- или pk-алгебру), включающую в себя почти все известные ранее системы открытого распределения ключей.

В последнее время наблюдается повышенный интерес исследователей к изучению математических моделей систем обработки и хранения дискретной информации (в том числе криптографических систем), представляющих собой алгебраические структуры – многоосновные универсальные алгебры (см., например, [2 – 6]). Отметим, что общая алгебраическая модель шифра по существу была предложена еще К. Шенноном [7]. Моделирование работы реальных устройств (алгоритмов) преобразования информации с помощью алгебраических структур дает возможность приблизиться к более глубокому пониманию принципов их построения и функционирования и позволяет на абстрактом уровне решать различные теоретические задачи анализа и синтеза таких устройств.

II Постановка задачи

Разнообразие известных в настоящее время систем открытого шифрования и цифровой подписи обусловливает актуальность задачи построения абстрактых алгебраических моделей указанных систем с целью анализа и осмысления с единых общих позиций их криптографических свойств, а также уточнения специфических особенностей, присущих конкретным криптосистемам открытого шифрования и цифоровой подписи. В отличие от известной модели системы открытого распределения ключей (pk-алгебры [2]), авторам не удалось найти в общедоступной литературе описание аналогичных моделей систем открытого шифрования и цифровой подписи, представленных в виде многоосновных универсальных алгебр.

III Основная часть

Как известно, существенной особенностью несимметричных криптосистем является использование различных ключей для зашифрования и расшифрования информации. При этом указанные ключи связаны определенной функциональной зависимостью таким образом, что вычислить по одному из них (открытому) второй (секретный) практически невозможно. Именно наличие такой зависимости позволяет организовать обмен шифрованными сообщениями с использованием только открытых каналов связи, то есть отказаться от секретных каналов для преварительного обмена ключами.

Один из возможных общих подходов к формальному определению системы открытого шифрования состоит в следующем. Рассмотрим многоосновную алгебру