УДК 004.9

А.Б. Сизоненко

Параллельная реализация криптографических блоков подстановок и перестановок арифметическими полиномами

Криптографические блоки подстановок и перестановок представлены в виде систем логических функций. Построены арифметико-логические модели блоков подстановок и перестановок. Рассмотрена возможность распараллеливания процесса вычисления функций криптографических блоков подстановок и перестановок за счет использования ресурсов арифметических вычислителей.

Ключевые слова: криптографические алгоритмы, подстановки, перестановки, арифметические полиномы, параллельные логические вычисления.

Задача разработки арифметико-логических моделей криптографических блоков подстановки и перестановки. Постановка задачи. В условиях противоборства в информационной сфере остро встает вопрос обеспечения защиты информации от несанкционированного доступа, подмены сообщений. Решить указанные задачи позволяет использование современных симметричных криптографических алгоритмов, таких как: ГОСТ 28147–89, DES, FEAL и др. [8], функциональными блоками большинства которых являются блоки перестановок и подстановок. Выполнение криптографических преобразований связано с выполнением интенсивных логических вычислений и действиями, связанными с выборкой из памяти. Таким образом, для выполнения криптографических преобразований используется ограниченный набор команд.

Для повышения производительности за счет распараллеливания процесса вычисления современные ЭВМ, как правило, имеют несколько вычислителей [5, 7]. Возможна разнесенная архитектура процессора, когда процессор состоит из двух связанных подпроцессоров, каждый из которых выполняет определенный набор команд [5]. В основном второй подпроцессор ориентирован на выполнение арифметических операций. В суперскалярных микропроцессорах и микропроцессорах с длинным командным словом уже содержится несколько вычислителей, каждый из которых предназначен для выполнения определенного набора команд. Кроме того, известны способы использования графических процессоров [6], которые имеются в любой ЭВМ, для решения задач общего назначения. Такие процессоры также ориентированы на выполнение арифметических операций.

Таким образом, в ЭВМ имеется ряд вычислителей, которые ориентированы на выполнение арифметических операций и при выполнении криптографических преобразований задействованы слабо либо вовсе не задействованы. Возникает задача использования вычислительной мощности этих вычислителей для выполнения криптографических преобразований. Для решения этой задачи необходимо представить логические функции блоков подстановок и перестановок с использованием таких операций, которые бы могли выполняться указанными выше вычислителями.

Для этого можно использовать математический аппарат представления систем булевых функций арифметическими полиномами. Процесс вычисления значений блоков подстановок и перестановок, представленных арифметическим полиномом, более трудоемкий, чем обращение к таблицам замен или перестановка бит. Однако такое представление дает возможность расширить набор команд за счет использования в этих целях арифметических операций и распараллелить процесс криптографических преобразований, путем задействования для выполнения логических операций ресурсов имеющихся в составе ЭВМ вычислителей, ориентированных на выполнение арифметических операций.

Общие сведения о криптографических блоках подстановки и перестановки. Подстановка – это операция, изменяющая порядок элементов в перестановке [4]. Подстановка приводит к замене каждого элемента исходной перестановки некоторым другим элементом.

Для задания подстановки необходимо задать всю совокупность элементов, над которыми производятся подстановка и алгоритм подстановки [4].

Взаимно однозначная функция $f: X \to X$ называется подстановкой на X [2]. Если множество X конечно, то можно считать, что $X = 1,\ldots,n$. В этом случае подстановку $f: 1,\ldots,n \to 1,\ldots,n$ удобно задавать таблицей из двух строк:

$$\begin{pmatrix} p_1\, p_2 \cdots p_n \\ q_1\, q_2 \cdots q_n \end{pmatrix}. \qquad (1)$$

В первой строке выражения (1) задаются значения аргументов, во второй – соответствующие значения функции. В таблице подстановки нижняя строка (значение функции) является перестановкой элементов верхней строки (значения аргумента). Если элементы верхней строки (аргументы) всегда располагаются в определенном порядке, то верхнюю строку можно не указывать – подстановка определяется одной нижней строкой.

В блочных криптоалгоритмах используются блоки нелинейной замены (*S*-блоки) (рис. 1), реализующие подстановку и имеющие n входов и d выходов ($n \geq d$).

Входы разделены на две группы. На входы $x_1,\ldots,x_d$ подается значение аргумента подстановки. Значения на входах $x_{d+1},\ldots,x_n$ определяют правило подстановки.

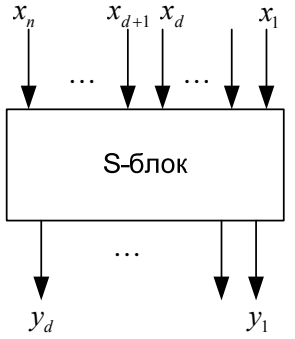

Рис. 1. Блок подстановок

Блок нелинейной замены задается матрицей размером 2^d столбцов и 2^{n-d} строк:

$$\mathbf{V}_S = \begin{bmatrix} v_{11} & v_{12} & v_{12^d} \\ v_{21} & v_{22} & v_{22^d} \\ \\ v_{2^{n-d}1} & v_{2^{n-d}2} & v_{2^{n-d}2^d} \end{bmatrix},\ v \in \{0,\ldots,2^d - 1\}. \qquad (2)$$

Значение на выходе блока нелинейной замены равно элементу матрицы $\mathbf{V}_S$ (2), порядковый номер строки которого равен значению на входах $x_{d+1},\ldots,x_n$, а номер столбца равен значению, подаваемому на входы $x_1,\ldots,x_d$. Значение на выходе *S*-блока определяется по формуле

$$Y = y_1 * y_2 \ldots * y_d = v_{KX},$$

где $Y = \sum_{i=0}^{d-1} y_{i+1} \cdot 2^i$; $X = \sum_{i=0}^{d-1} x_{i+1} \cdot 2^i$; $K = \sum_{i=1}^{n-d} x_{i+d} \cdot 2^{i-1}$.

Если правило подстановки применить к индексам входных переменных, то получим блок перестановок. Аппаратно блок перестановок реализуется с помощью коммутатора. В существующих криптоалгоритмах коммутаторы могут иметь *n* входов и *m* выходов ($m \neq n$). Функция перестановки задается вектором:

$$\mathbf{P} = \begin{bmatrix} p_0 & \ldots & p_i & \ldots & p_{m-1} \end{bmatrix}, \qquad (3)$$

где $p_i \in (0, n-1)$ – определяет, на какой выход коммутируется переменная с входа *i*.

Представление булевых функций арифметическими полиномами. Произвольный кортеж булевых функций $f_d(X) * f_{d-1}(X) * \cdots * f_1(X)$ может быть единственным образом представлен арифметическим полиномом [1]:

$$Y = D(X) = \sum_{i=0}^{2^n-1} c_i x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n},$$

где $\mathbf{C} = (c_0 \ldots c_{2^n-1})$ – целочисленный вектор коэффициентов арифметического полинома; $i = (i_{n-1} i_{n-2} \ldots i_0) = \sum_{u=0}^{n-1} i_u 2^u$, $i_u \in \{0,1\}$ – разряды двоичной системы счисления.

Арифметический полином, описывающий систему логических функций, можно получить алгебраическим и матричным способами [1, 3]. Алгебраический способ заключается в реализации следующего алгоритма.

Алгоритм 1 [1, 3]:

Шаг 1. Получение арифметических полиномов $P_i(X)$ для каждой булевой функции $y_j = f_j(X)$, $j = 1,\ldots,d$, по формулам замены логических операций на арифметические: $x \oplus y = x + y - 2xy$; $x \vee y = x + y - xy$; $x \wedge y = xy$; $\bar{x} = 1 - x$.

Шаг 2. Получение арифметических полиномов, взвешенных весами 2^{j-1} $(j = 1,\ldots,d)$.

Шаг 3. Получение искомого арифметического полинома $D(X)$ путем суммирования арифметических полиномов, полученных в шаге 2, и приведение подобных слагаемых.

Матричное преобразование выполняется следующим образом [1, 3]:

$$\mathbf{C} = \mathbf{A}_{2^n} \cdot \mathbf{Y}, \tag{4}$$

где $\mathbf{Y}$ – вектор истинности булевой функции; $\mathbf{A}_{2^n}$ – матрица прямого арифметического преобразования размерности $2^n \times 2^n$. Матрица $\mathbf{A}_{2^n} = \begin{bmatrix} \mathbf{A}_{2^n-1} & 0 \\ -\mathbf{A}_{2^n-1} & \mathbf{A}_{2^n-1} \end{bmatrix}$ называется n-й кронекеровской степенью $\mathbf{A}_{2^n} = \bigotimes_{j=1}^{n} \mathbf{A}_1$ базовой матрицы $\mathbf{A}_1 = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$.

Матричные преобразования хорошо алгоритмизируются и удобны для практического применения.

Если $m < Y_{\max}$, где $Y_{\max}$ – максимальное значение, принимаемое Y, то произвольный кортеж логических функций может быть представлен модулярным арифметическим полиномом [3]:

$$Y = MD(X) = \left| \sum_{i=0}^{2^n-1} \psi_i (x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}) \right|_m^+,$$

где $\psi_i = |c_i|_m^+$. Коэффициенты модулярного арифметического полинома $MD(X)$ лежат в области целых неотрицательных чисел, а их числовой диапазон равен значению модуля m [3].

Алгоритм представления криптографических блоков подстановки и перестановки арифметическими полиномами. В соответствии с вектором (3), задающим блок перестановок, получим систему выражений, описывающих этот блок:

$$\begin{cases} x'_0 = x_{p_0}, \\ \vdots \\ x'_i = x_{p_i}, \\ \vdots \\ x'_{m-1} = x_{p_{m-1}}, \end{cases} \tag{5}$$

где x'_i – значение i-го выхода; x_{p_i} – значение p_i входа блока перестановок $i \in (0,\ldots,m-1)$.

Построим арифметическую модель по системе выражений (5). Для этого применим алгоритм получения арифметического полинома (алгоритм 1). Так как значения булевых функций заданы всего лишь одним значением переменной и будут иметь такое же арифметическое представление, то переходим сразу ко второму шагу – получение выражений, взвешенных весами 2^j $(j = 0,\ldots,m-1)$:

$$\begin{cases} x'_0 = x_{p_0}, \\ \vdots \\ x'_i = 2^i x_{p_i}, \\ \vdots \\ x'_{m-1} = 2^{m-1} x_{p_{m-1}}. \end{cases} \tag{6}$$

Далее, суммируем выражения, составляющие систему (6), и получаем полином:

$$DL^{(K)}(X) = \sum_{i=0}^{m-1} 2^i x'_i = \sum_{i=0}^{m-1} 2^i x_{p_i}. \tag{7}$$

Выражение (7) назовем арифметической моделью блока перестановок.

Пример 1.

Рассмотрим блок перестановок, имеющий пять входов и пять выходов и заданный вектором

$$\mathbf{P}=\begin{bmatrix}3 & 2 & 0 & 4 & 1\end{bmatrix}.$$

Аппаратно этот блок будет реализован с помощью коммутатора, представленного на рис. 2.

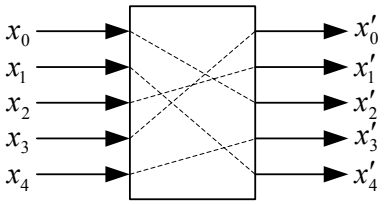

Рис. 2. Схема коммутатора

В соответствии с (6) получим АП, описывающий заданный блок перестановок:

$$D^{(K)}(X)=2^0x_3+2^1x_2+2^2x_0+2^3x_4+2^4x_1. \tag{8}$$

При подаче на вход коммутатора значений $(x_4x_3x_2x_1x_0)=(11010)$ в соответствии с (8) получим

$$D^{(K)}(X)=2^0\cdot1+2^1\cdot0+2^2\cdot0+2^3\cdot1+2^4\cdot1=(25)_{10}=(11001)_2.$$

Получение арифметической модели блока подстановок заключается в выполнении следующего алгоритма.

Алгоритм 2:

Шаг 1. Получение целочисленного задания системы булевых функций, описывающей блок подстановок.

Блок подстановок представляет собой цифровое комбинационное устройство с n входами и d выходами. Блок подстановок, заданный матрицей (2), можно задать вектором, расположив строки матрицы последовательно в одну строку. Таким образом, получим целочисленное задание системы булевых функций:

$$\mathbf{Y}=\begin{bmatrix}v_{11} & \dots & v_{12^d} & v_{21} & \dots & v_{22^d} & \dots & v_{2^{n-d}1} & \dots & v_{2^{n-d}2^d}\end{bmatrix}.$$

Шаг 2. Получение коэффициентов арифметического полинома, описывающего блок подстановок.

Для получения арифметического полинома применим прямое матричное преобразование (4). В результате получим вектор $\mathbf{C}=\begin{bmatrix}c_0 & \dots & c_{2^n-1}\end{bmatrix}$, определяющий коэффициенты арифметического полинома.

Шаг 3. Сопоставив коэффициенты $\mathbf{C}$ членам арифметического полинома, получим арифметическую модель блока подстановок:

$$D^{(S)}(X)=\sum_{i=0}^{2^n-1}c_ix_n^{i_n}\dots x_1^{i_1}. \tag{9}$$

Шаг 4. Получение модулярной формы блока подстановки:

$$DM^{(S)}(X)=\left|\sum_{i=0}^{2^n-1}|c_i|^+_{2^d}x_n^{i_n}\dots x_1^{i_1}\right|^+_{2^d}. \tag{10}$$

Пример 2.

Рассмотрим S-блок, заданный вектором $\mathbf{Y}=\begin{bmatrix}5 & 2 & 7 & 0 & 3 & 1 & 4 & 6\end{bmatrix}$.

Данный S-блок содержит три входа и три выхода. Для получения вектора коэффициентов арифметического полинома применим прямое матричное преобразование:

$$C=\begin{bmatrix}1&0&0&0&0&0&0&0\\-1&1&0&0&0&0&0&0\\-1&0&1&0&0&0&0&0\\1&-1&-1&1&0&0&0&0\\-1&0&0&0&1&0&0&0\\1&-1&0&0&-1&1&0&0\\1&0&-1&0&-1&0&1&0\\-1&1&1&-1&1&-1&-1&1\end{bmatrix}\cdot\begin{bmatrix}5\\2\\7\\0\\3\\1\\4\\6\end{bmatrix}=\begin{bmatrix}5\\-3\\2\\-4\\-2\\1\\-1\\8\end{bmatrix}\begin{matrix}\\x_0\\x_1\\x_1x_0\\x_2\\x_2x_0\\x_2x_1\\x_2x_1x_0\end{matrix}.$$

Подставим значения коэффициентов в выражение (9), получим арифметическую модель S-блока:

$$D^{(S)}=5-3x_0+2x_1-4x_1x_0-2x_2+1x_2x_0-1x_2x_1+8x_2x_1x_0. \tag{11}$$

При поступлении на вход S-блока комбинации (101), в соответствии с (11) получим $D^{(S)}=1$, что соответствует вектору, задающему S-блок.

Для получения модулярной формы применим формулу (10). Так как количество выходов равно трем, то значение модуля примем равным 2^3.

$$D^{(S)} = \left|5 + 5x_0 + 2x_1 + 4x_1x_0 + 6x_2 + 1x_2x_0 + 7x_2x_1\right|_8^+ . \quad (12)$$

При поступлении на вход S-блока той же комбинации (101), в соответствии с (12) также получим $D^{(S)} = |17|_8^+ = 1$.

Заключение. Полученные результаты позволяют сделать следующие выводы:

1. Блок перестановок может быть задан линейным арифметическим полиномом.

2. Блок подстановок (*S*-блок) задается нелинейным арифметическим полиномом. Модулярная форма позволяет уменьшить количество членов арифметического полинома.

Таким образом, полученные алгоритмы можно использовать для распараллеливания процесса вычислений функций, реализуемых блоками подстановок и перестановок, путем передачи части вычислений устройствам, ориентированным на выполнение арифметических операций.

Литература

1. Малюгин В.Д. Параллельные логические вычисления посредством арифметических полиномов. – М.: Наука. Физматлит, 1997. – 192 с.

2. Новиков Ф. А. Дискретная математика для программистов. – СПб.: Питер, 2002. – 304 с.

3. Финько О.А. Модулярная арифметика параллельных логических вычислений / Под ред. В.Д. Малюгина. – М.: Институт проблем управления им. В.А. Трапезникова РАН; Краснодар: Краснодарский военный институт, 2003. – 224 с.

4. Фрид Э. Элементарное введение в абстрактную алгебру / пер. с венгер. Ю.А. Данилова. – М.: Мир, 1979. – 260 с.

5. Корнеев В.В. Современные микропроцессоры / В.В. Корнеев, А.В. Киселев. – 3-е изд. перераб. и доп. – СПб.: БХВ-Петербург, 2003. – 448 с.

6. Линев А.В. Технологии параллельного программирования для процессоров новых архитектур: учеб. / А.В. Линев, Д.К. Боголепов, С.И. Бастраков; под ред. В.П. Гергеля. – М.: Изд-во Моск. ун-та, 2010. – 160 с.

7. Гергель В.П. Высокопроизводительные вычисления для многопроцессорных и многоядерных систем: учеб. – М.: Изд-во Моск. ун-та, 2010. – 544 с.

8. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. – М.: ТРИУМФ, 2003. – 816 с.

Сизоненко Александр Борисович
Зам. нач. каф. оперативно-разыскной деятельности и специальной техники
Краснодарского университета МВД России
Тел.: 8 (861) 258-36-81
Эл. почта: siz_al@mail.ru

Sizonenko A.B.

Parallel implementation of cryptographic blocks of substitution and transposition by arithmetic polynomials

Cryptographic blocks of substitution and transposition are presented as systems of logical functions. Arithmetic-logic model of blocks of substitution and transposition are constructed. The possibility of parallelizing the computation of the cryptographic functions of the blocks of substitutions and permutations by using the resources of arithmetic calculators is considered.

Keywords: cryptographic algorithms, substitution, transposition, arithmetic polynomials, parallel logical computation.