Оптимизация вычислений

Рассматривается симметричный блочный криптоалгоритм WBC1 и его параллельная реализация – криптоалгоритм PWBC1.

УДК 519.6

И.А. БАРАНОВ

ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ PWBC1

Введение. В современной криптографии сложилась тенденция, что, шифруемые данные представлены на листе бумаге, т. е. на плоскости [1]. Для шифрования данных осуществляются всевозможные действия над матрицами значений или цепочкой символов. Для усложнения алгоритмов вводятся более сложные операции. Для обработки данные разбиваются на большие блоки.

Уникальность и нововведение в алгоритме WBC1, отличающие его от других известных алгоритмов, заключается в том, что в этом криптоалгоритме данные представлены в трехмерном пространстве. В основе алгоритма заложен известный всем принцип Кубика – Рубика. Используя этот метод, мощность алгоритма увеличивается на несколько порядков. А как известно мощность алгоритма – это стойкость к криптоанализу и вскрытию в лоб [2], т. е. полному перебору.

В работе рассматривается модификация алгоритма WBC1 для параллельных компьютеров MIMD-архитектуры.

1. Алгоритм WBC1. Первично был разработан алгоритм последовательной обработки [3]. WBC1 представляет собой блочный шифр, шифрующий данные 32, 64, 216 и 512-битовыми блоками. С одного конца алгоритма вводится 64, 216 или 512-битовый блок открытого текста, а с другого конца выходит 32, 64, 216 или 512-битовый блок шифротекста.

WBC1 является симметричным алгоритмом: для шифрования и дешифрования используется одинаковый алгоритм и ключ. Длина ключа должна быть не менее 64 бит. Ключ, который может быть любым 64-битным числом, можно изменить в любой момент времени.

Криптостойкость полностью определяется ключом. Фундаментальным строительным блоком WBC1 является комбинация вертикальных и горизонтальных перестановок, количество которых прямо пропорционально зависит от длины и элементов ключа. Так, если размер блока, на которые разбиваются входные данные, равен 32 битам, то количество всех возможных состояний будет равно $(2^{15})!$, что соответствует стойкости алгоритма.

Процесс шифрования заключается в том, что шифруемые данные разбиваются на блоки (рис. 1) и записываются в трехмерный массив (в куб), минимальный блок 32 бита.

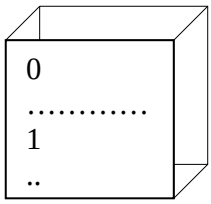

РИС. 1. Шифруемый блок

Подобно алгоритму DES создается таблица перестановок, в которую записываются 127 всевозможных не соответствующих одна другой операций над кубом (т.е. оборотов его плоскостей и комбинаций). После предварительных установок этапы шифрования проходят в четкой последовательности.

Последовательное сканирование элементов ключа. В зависимости от значения элемента ключа применяется тот или иной поворот плоскостей куба из возможных 127 комбинаций.

После каждого прохода отдельного символа ключа реализуется циклический побитовый сдвиг (рис. 2) (тип циклического сдвига зависит от реализации алгоритма, т. е. от мощности).

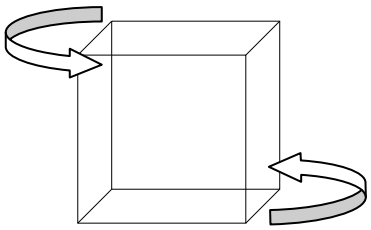

РИС. 2. Циклический побитовый сдвиг в блоке

Когда будет завершен проход всего ключа, данные расформировываются в цепочку символов.

Процесс дешифрования заключается в обратном проходе всех операций шифрования.

В общем виде цикл преобразований представлен на рис. 3.

Учитывая что для шифрования и дешифрования алгоритмом WBC1 используются только операции перестановки из массива в массив и циклический побитовый сдвиг, затраты памяти идут только на создание и работу с двумя трехмерными массивами и битовую операцию. Можно заметить, что уязвимым местом алгоритма является ограниченное количество перестановок (127), но операция циклического побитового сдвига решает эту проблему.

Мощность алгоритма может увеличиваться в зависимости от поставленной задачи. Например, для шифрования документов на домашнем компьютере можно ограничиться минимальными возможностями, такими как длина ключа – до 80 бит и размер обрабатываемых блоков – 64 бита. Но для шифрования документов в государственных организациях этого уже не хватит и следует использовать более длинные ключи (с дополнительным процессом расширения ключа). Возможности алгоритма позволяют реализовывать его как на программном уровне, так и на аппаратном.

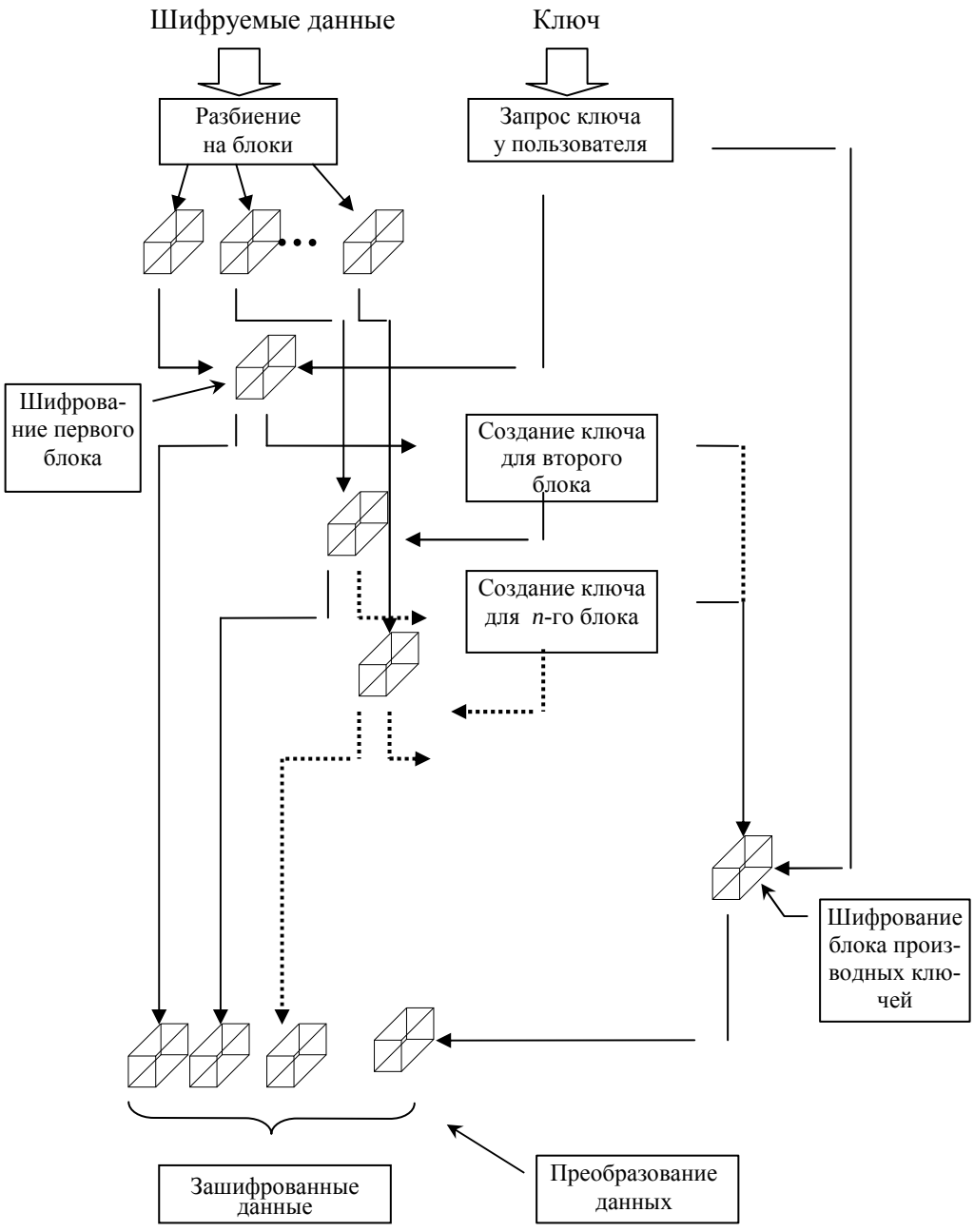

РИС. 3. Схема выполнения алгоритма WBC1

2. Параллельный алгоритм PWBC1. Применяя методы параллелизма при реализации криптографических алгоритмов можно достичь определенных выгод, но основным критерием, влияющим на эффективность, будет оставаться форма реализации алгоритма.

На основе предложенного выше алгоритма WBC1, разработан параллельный блочный криптографический алгоритм PWBC1. Отличие от алгоритма WBC1 состоит в возможности параллельной обработки кубических матриц данных независимо друг от друга на разных процессорах (ядрах). Также пришлось внести изменения в процесс формирования ключа.

При добавлении возможности независимой обработки шифруемых блоков существенно ускорилась обработка (шифрование) шифруемого текста, при этом появилась возможность увеличить длину ключа шифрования.

В алгоритме PWBC1, в отличие от алгоритма WBC1, в котором ключ для каждого блока формировался из предыдущего блока и части ключа, ключ формируется из начального ключа к которому применяются специальные методы. В связи с этим каждый блок не зависит от соседнего блока, что в свою очередь повышает криптостойкость алгоритма.

PWBC1 является симметричным алгоритмом: для шифрования и дешифрования используется одинаковый алгоритм и ключ. Длина ключа должна быть не менее 128 бит.

Криптостойкость полностью определяется ключом. Фундаментальным строительным блоком PWBC1 является комбинация вертикальных и горизонтальных перестановок, количество которых прямо пропорционально зависит от длины и элементов ключа.

Процесс шифрования заключается в том, что шифруемые данные разбиваются на блоки (см. рис. 1) и записываются в трехмерный массив (в куб), минимальный блок 32 бита. Для ускорения процесса шифрования используются принципы параллелизма. Каждый из блоков обрабатывается на своем процессоре (ядре). В общем виде цикл преобразований представлен на рис. 4.

После предварительных установок этапы шифрования проходят в четкой последовательности, каждый блок распределяется по процессорам, что дает возможность распараллелить алгоритм.

1. Последовательное сканирование элементов ключа. В зависимости от значения элемента ключа применяется тот или иной поворот плоскостей куба из возможных 127 комбинаций.

2. После каждого прохода отдельного символа ключа реализуется циклический побитовый сдвиг (тип циклического сдвига зависит от реализации алгоритма, т. е. от мощности).

3. Когда будет завершен проход всего ключа, данные расформировываются в цепочку символов.

Процесс дешифрования заключается в обратном прохождении всех операций шифрования.

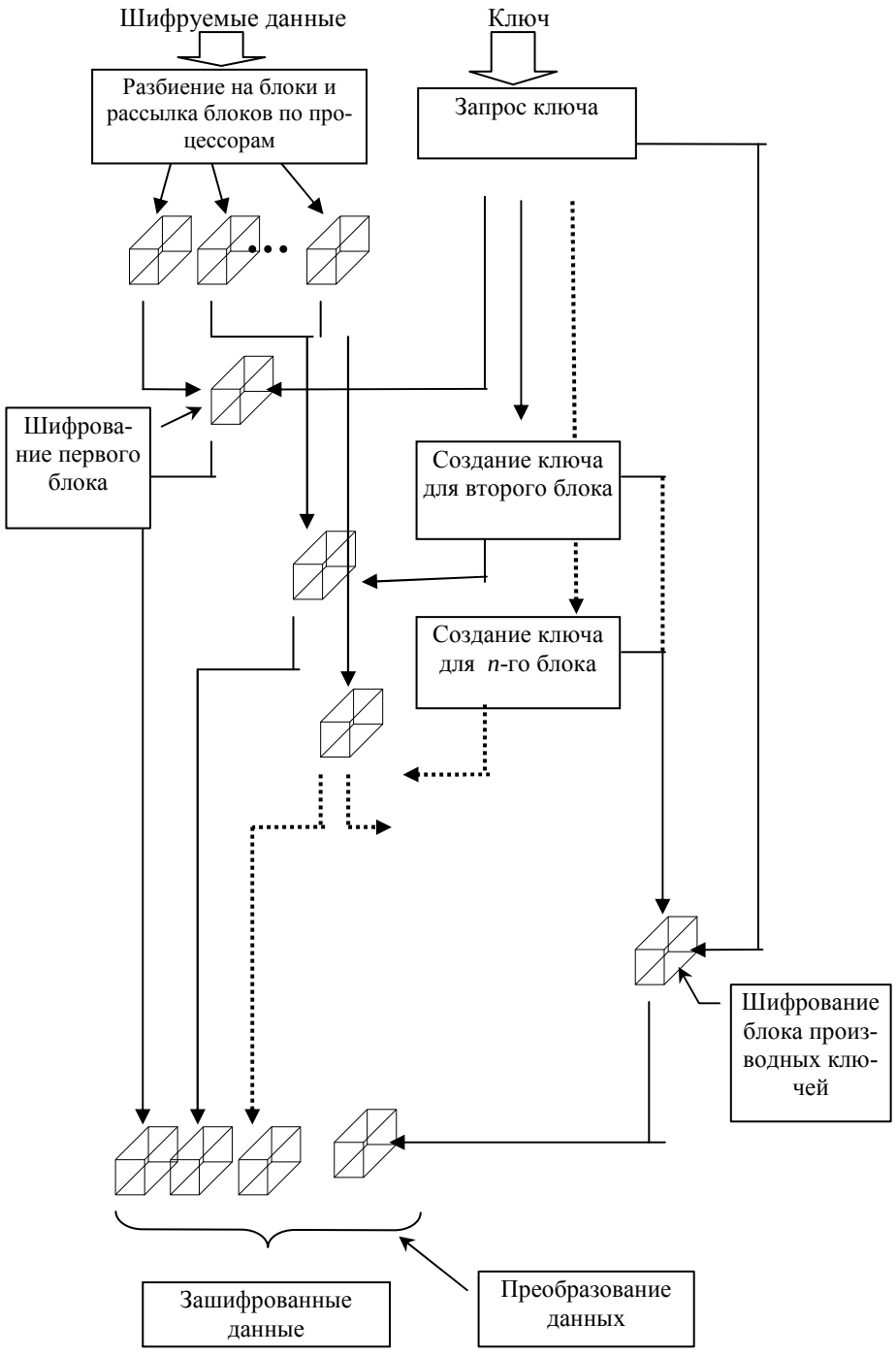

РИС. 4. Схема выполнения алгоритма PWBC1

Заключение. Используя алгоритм PWBC1 на многопроцессорном (ядерном) компьютере можно достигнуть более эффективных результатов как по времени шифрования так и по стойкости алгоритма. Предложенный параллельный алгоритм более эффективен по скорости обработки данных. Если пренебречь временем на пересылку данных между процессорами, то увеличение скорости выполнения алгоритма пропорционально количеству процессоров.

І.А. Баранов

ПАРАЛЕЛЬНИЙ АЛГОРИТМ PWBC1

Розглядається симетричний блочний криптоалгоритм WBC1 та його паралельна реалізація – криптоалгоритм PWBC1.

I.A. Baranov

PWBC1 PARALLEL ALGORITHM

Symmetrical block WBC1 cryptoalgorithm and its parallel implementation – PWBC1 cryptoalgorithm - are considered.

1. *Глушков В.М.* Основы безбумажной информатики. – М.: Наука, 1982. – 552 с.
2. *Михалевич В.С., Сергиенко И.В., Шор Н.З.* Исследование алгоритмов решения оптимизационных задач и их приложения // Кибернетика. – 1981. – № 4. – С. 89 – 113.
3. *Баранов І.А.* Алгоритм WBC1 // Питання оптимізації обчислень (ПОО-XXXV): Міжнародний симпозіум, 24 – 29 вересня, 2009. – Київ. – 2009. – С. 47.

Получено 30.03.2010

Об авторе:

Баранов Игорь Анатольевич,
ведущий инженер-программист Института кибернетики имени В.М. Глушкова НАН Украины.
e-mail: vlasov@ukr.net