8. Мандрикова О. В., Заляев Т. Л. Моделирование вариаций космических лучей на основе совмещения кратномасштабных вейвлет-разложений и нейронных сетей переменной структуры // Цифровая обработка сигналов. 2015. № 1. С. 11–16.

9. Kozlov V. I. Markov V. V. Wavelet image of a heliospheric storm in cosmic rays // Geomagnetism and Aeronomy. 2007. Vol. 47, № 1. P. 56–65.

10. Kozlov V. I., Kozlov V. V. A new index of solar activity: An index of cosmic ray scintillation // Geomagnetism and Aeronomy. 2008. Vol. 48, № 4. P. 1–9.

11. Mandrikova O. V., Solovev I. S.. Zalyaev T. L. Methods of analysis of geomagnetic field variations and cosmic ray data // Earth Plan. Space. 2014. 66:148, doi:10. 1186/s40623-014-0148-0.

12. Mallat S. A Wavelet Tour of Signal Processing, Third Edition:The Sparse Way. USA: Academic Press, 2008. 832 p.

13. Левин Б. Р. Теоретические основы статистической радиотехники. 2-е изд. М.: Сов. радио, 1975. 392 с.

14. Ruffolo D. Transport and acceleration of energetic charged particles near an oblique shock // Astrophys. J. 1999. № 515. P. 787–800.

15. Cosmic ray anisotropy before and during the passage of major solar wind disturbances / A. V. Belov, J. W. Bieber, E. A. Eroshenko, P. Evenson, R. Pyle, V. G. Yanke // Adv. Space Res. 2003. Vol. 31, № 4. P. 919–924.

T. L. Zalyaev
Institute of Cosmophysical Research and Radio Wave Propagation of the Far Eastern Branch of Russian Academy of Science

ALGORITHM FOR ANOMALY DETECTION IN COSMIC RAYS VARIATIONS IN PERIODS OF HELIOSPHERIC DISTURBANCES

In this paper author proposed an algorithm for detailed analysis of the variations of cosmic rays and allocation of local abnormal changes that occur during periods of heliospheric disturbances. The algorithm is based on a combination of wavelet transform and threshold functions. The efficiency of the algorithm In this paper is shown on the basis of neutron monitors data (author analyzed data from neutron monitor stations Apatity and Cape Schmidt). The analysis of the variations of cosmic rays during periods of increased solar activity was performed and abnormal changes that occur before and during the moments of strong geomagnetic disturbances were highlighted.

Wavelet transform, cosmic rays, heliospheric disturbances, forbush effect, magnetic storms

УДК 004.056.55

Ю. О. Чернышев, А. С. Сергеев, А. Н. Рязанов
Донской государственный технический университет

С. А. Капустин
Краснодарское высшее военное училище им. генерала армии С. М. Штеменко

Исследование возможности применения методов эволюционной оптимизации для реализации криптоанализа блочных методов шифрования

Рассматривается задача криптоанализа методов криптографической защиты с использованием новой модели оптимизационных методов – генетических алгоритмов, имитирующих процессы эволюции живой природы. Описывается применение генетических алгоритмов для криптоанализа блочного стандарта шифрования России. Приводятся структурная схема генетического алгоритма, описание и отличительные особенности процесса криптоанализа, результаты экспериментальной реализации.

Криптоанализ, биоинспирированные методы, генетический алгоритм, блочный алгоритм шифрования, популяция ключей, кроссинговер, квазиоптимальный ключ

В настоящее время при разработке компьютерных технологий, обеспечивающих информационную безопасность и защиту информации, широкое применение находят криптографические методы защиты. Для решения задач криптоанализа, относящихся к классу NP-полных, в послед-

ние годы применяются алгоритмы, основанные на природных системах. К ним относятся методы моделирования отжига, генетические алгоритмы (ГА), эволюционные методы, алгоритмы роевого интеллекта и т. д. В моделях и алгоритмах эволюционных вычислений ключевым элементом является построение начальной модели и правил, по которым она может изменяться (эволюционировать). В течение последних лет были предложены разнообразные схемы эволюционных вычислений, в том числе генетический алгоритм, генетическое программирование, эволюционные стратегии, эволюционное программирование.

В работе [1] рассматривались задачи криптоанализа и приведены результаты криптоанализа классических симметричных криптографических алгоритмов с использованием методов эволюционной оптимизации и генетического поиска для симметричных шифров перестановок, а также для реализации шифров простой и многоалфавитной замены. Среди обзорных работ, посвященных описанию методов и перспектив развития криптоанализа, следует отметить [2]–[4], в которых приведены универсальные методы (метод полного перебора, атака по ключам, частотный анализ, метод Полларда), методы криптоанализа симметричных (статистический метод, метод дифференциального анализа, метод линейного анализа) и асимметричных (задача дискретного логарифмирования, задача факторизации) криптосистем, а также новый вид криптоанализа – атаки по побочным каналам. В работе [2] также приводится краткое изложение новых технологий, связанных с использованием ГА, нейронных сетей и квантовых компьютеров.

Таким образом, возникает вопрос о возможности применения биоинспирированных методов для криптоанализа современных блочных алгоритмов шифрования, так как переход к блочному шифрованию открывает дополнительные возможности для повышения стойкости криптоалгоритмов. Основные принципы построения блочных шифров, структура алгоритмов блочного шифрования (схема Фейстеля) описаны, например, в [3].

Следует отметить, что отличительной особенностью применения биоинспирированных методов криптоанализа (в частности, ГА) является возможность использования самого алгоритма шифрования (или расшифровки) в качестве целевой функции для оценки пригодности ключа, определенного с помощью генетических операций. В связи с этим можно утверждать, что при использовании ГА процесс определения секретного ключа (например, при криптоанализе 2-го типа) зависит не столько от сложности шифрующих преобразований, сколько от самого биоинспирированного метода, который должен обеспечивать достаточное разнообразие генерации ключей. Таким образом, задача исследования возможности применения биоинспирированных алгоритмов для криптоанализа блочных криптосистем является, несомненно, актуальной.

Реализация криптоанализа блочных методов с использованием ГА на примере представителя блочных шифров – стандарта DES наряду с экспериментальными результатами была представлена в [5], [6]. Аналогичным образом опишем применение генетических методов для организации криптоанализа стандарта шифрования ГОСТ 28147–89 [7].

Заметим, что важным свойством как блочных методов, так и ГА является их внутренний параллелизм. Основные модели параллельных ГА (глобальный параллельный ГА, островная модель, клеточный ГА) приведены в [1]. Для разработки криптоанализа алгоритма с помощью эволюционного подхода рассмотрим вначале процесс параллельной реализации составляющих его этапов. Работа алгоритма с введенными терминами и обозначениями, используемыми далее, описана в [8]. В соответствии с этим алгоритм предусматривает 4 режима работы:

– шифрование данных в режиме простой замены;

– шифрование данных в режиме гаммирования;

– шифрование данных в режиме гаммирования с обратной связью;

– выработка имитовставки.

В режиме простой замены на глобальном уровне в соответствии со структурной схемой, приведенной в [8], можно выделить следующие параллельно выполняемые этапы:

– параллельную обработку 64-битовых блоков исходного текста;

– параллельную обработку 32-разрядного вектора в восьми узлах замены.

С учетом этих очевидных преобразований структурная схема одного цикла режима простой замены представлена на рис. 1.

Секретность в режиме простой замены определяется как 256-битовым ключом, так и восемью узлами замены, каждый из которых содержит 16 четырехразрядных двоичных чисел.

В режиме гаммирования шифрование блока исходного текста осуществляется поразрядным суммированием с блоком гаммы шифра, вырабатываемой с помощью исходной синхропосылки $\hat{S}$ и 32-разрядных двоичных констант C_1 и C_2. Наряду с этим для шифрования в режиме простой замены также используется 256-битовый ключ.

Отметим, что структурные схемы шифрования в режиме простой замены и в режиме гаммирования представлены в [7]. Для данной структурной схемы (рис. 1) составим информационно-логическую граф-схему, отобразив на ней связи по управлению (жирная линия) и по информации (тонкая линия) (рис. 2). На рис. 2 жирной линией отмечены связи 15–16, 15–17 и 2–3, 2–4, ..., 2–10. (Для сокращения размеров рисунков и матриц элементы схемы на рис. 1, представляющие начальное разбиение текста на 64-битовые блоки и начальное формирование ключа, не пронумерованы и соответствующие вершины граф-схемы не показаны).

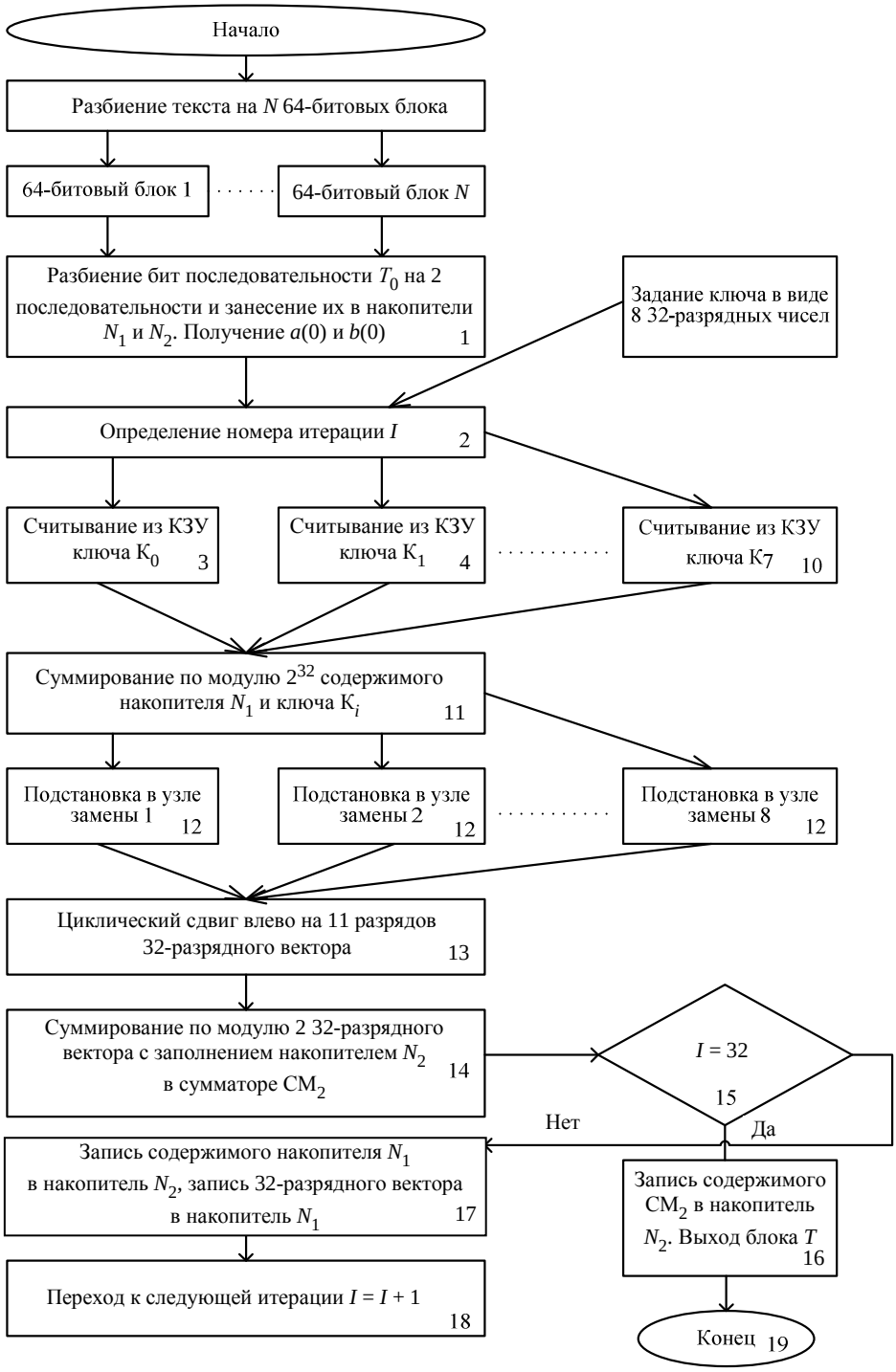

Рис. 1

Для данного графа введем в рассмотрение матрицу следования *S*. В соответствии с [9] элемент S_{ij} = *, если существует связь по управлению, и S_{ij} = = 1, если существует связь по информации.

Далее, используя алгоритмы, описанные в [9], дополним матрицу *S* транзитивными связями (табл. 1), обозначив все элементы S_{ij} = 1. Затем сформируем симметричную матрицу следования (табл. 2) и введем в рассмотрение матрицу *L* логической несовместимости операторов, также дополнив ее транзитивными связями (табл. 3). Дизъюнктивным сложением этих матриц получим матрицу независимости *M* (табл. 4).

Очевидно, что в общем случае применение ГА возможно для организации криптоаналитической атаки 2-го типа, т. е. при наличии известного текста и шифртекста требуется определить секретный 256-битовый ключ (с целью дешифрования других сообщений, зашифрованных тем же ключом), либо 64-битовую синхропосылку $\hat{S}$, используемую для генерации гаммы шифра (при известном секретном ключе и таблицах перестановки в режиме гаммирования), либо таблицы перестановки (при известном секретном ключе и синхропосылке). Таким образом, реализация ГА заключается в генерации популяции секретных ключей (синхропосылок или таблиц перестановок) и оценке их оптимальности с последующим применением стандартного набора генетических операций. Структурная схема, иллюстрирующая применение ГА для реализации криптографической атаки 2-го типа (при наличии 64-битовых блоков открытого текста и шифртекста) алгоритма DES приведена в [6].

В соответствии с данной схемой после формирования начальной популяции ключей производится оценка их пригодности, т. е. проверка, насколько полученный с их помощью шифртекст совпадает с заданным. После оценки производится селекция индивидуумов популяции для проведения множества генетических операций и получения множества потомков. Затем полученная расширенная популяция подвергается дальнейшему оцениванию. Процесс заканчивается либо когда прекращается эволюционирование популяции, либо когда исчерпан заданный временной ресурс (пройдено заданное количество генераций). Процесс криптоанализа алгоритма ГОСТ в общем случае может протекать аналогично.

Очевидно, что минимизация значения *T* времени работы ГА приводит к повышению быстродействия и эффективности ГА и возможна за счет распараллеливания процесса оценки элементов популяции как на глобальном уровне (параллельная обработка *P* элементов популяции на *n* процессорах), так и на локальном (параллельная реализация процесса оценки одного элемента популяции). Таким образом, для повышения эффективности реализации ГА на локальном уровне

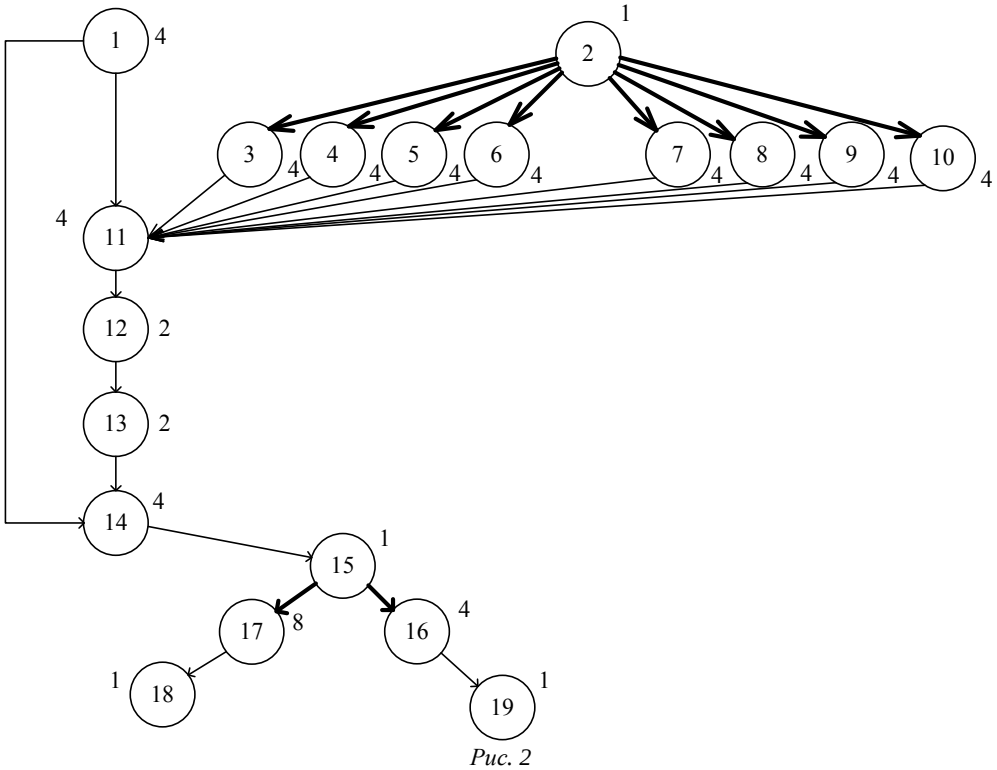

Рис. 2

необходимо определить минимальное число процессоров однородной вычислительной системы при заданных оценках времени выполнения операторов, составляющих граф-схему на рис. 2.

Для решения этой задачи воспользуемся методами, изложенными в [9].

Веса операторов, показывающие время их выполнения и определенные в соответствии с основными правилами анализа программ, описанными в [10], приведены на рис. 2. Они промасштабированы делением на 10 и округлены до ближайшего целого сверху. Легко убедиться, что критический путь в графе $T_{кр} = 27$, его образует последовательность операторов 2-3-11-12-13-14-15-17-18.

Далее примем заданное время $T_{зад} = T_{кр}$. Для представленного на рис. 2 информационно-логического графа и матрицы следования найдем ранние τ_{pi} и поздние сроки $\tau_{пi}$ окончания выполнения операторов с помощью алгоритмов, представленных в [9].

Ранние сроки:

$$\tau_{p1} = 4, \tau_{p2} = 1, \tau_{p3} = \tau_{p4} = \tau_{p5} = \tau_{p6} = \tau_{p7} = \tau_{p8} = \tau_{p9} = \\ = \tau_{p10} = 5, \tau_{p11} = 9, \tau_{p12} = 11, \tau_{p13} = 13, \tau_{p14} = 17, \\ \tau_{p15} = 18, \tau_{p16} = 22, \tau_{p17} = 26, \tau_{p18} = 27, \tau_{p19} = 23.$$

Поздние сроки:

$$\tau_{п19} = \tau_{п18} = 27, \tau_{п17} = \tau_{п16} = 26, \tau_{п15} = 18, \tau_{п14} = \\ = 17, \tau_{п13} = 13, \tau_{п12} = 11, \tau_{п11} = 9, \tau_{п10} = \tau_{п9} =$$

Таблица 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1																			
2																			
3		1																	
4		1																	
5		1																	
6		1																	
7		1																	
8		1																	
9		1																	
10		1																	
11	1	1	1	1	1	1	1	1	1	1									
12	1	1	1	1	1	1	1	1	1	1	1								
13	1	1	1	1	1	1	1	1	1	1	1	1							
14	1	1	1	1	1	1	1	1	1	1	1	1	1						
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1					
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1				
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1				
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1		
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			

Таблица 2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1											1	1	1	1	1	1	1	1	1
2			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3		1									1	1	1	1	1	1	1	1	1
4		1									1	1	1	1	1	1	1	1	1
5		1									1	1	1	1	1	1	1	1	1
6		1									1	1	1	1	1	1	1	1	1
7		1									1	1	1	1	1	1	1	1	1
8		1									1	1	1	1	1	1	1	1	1
9		1									1	1	1	1	1	1	1	1	1
10		1									1	1	1	1	1	1	1	1	1
11	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1	1
12	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
13	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1				1
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			1	
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1		
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			

$$= \tau_{п8} = \tau_{п7} = \tau_{п6} = \tau_{п5} = \tau_{п4} = \tau_{п3} = 5,$$
$$\tau_{п2} = 1, \tau_{п1} = 5.$$

Используя матрицу независимости, найдем внутренне устойчивые множества, представляющие множества взаимно независимых операторов (ВНО). Это множества (1,2), (1,3), (1,4), (1,5), (1,6), (1,7), (1,8), (1,9), (1,10).

Используя значения $\tau_{рi}$ и $\tau_{пi}$, оценим минимальное число процессоров для выполнения алгоритма за время $T_{кр}$ в соответствии с визуальной методикой, описанной в [9], т. е. строя диаграммы ранних и поздних сроков окончания выполнения операторов и находя такое распределение временных границ операторов, при котором число используемых процессоров минимально.

Легко убедиться, что максимальное значение $n = 2$ соответствует ВНО (1,3), ВНО (1,4), ВНО (1,5), ВНО (1,6), ВНО (1,7), ВНО (1,8), ВНО (1,9), ВНО (1,10), поскольку операторы, входящие в данные множества ВНО, имеют равные ранние и поздние сроки окончания выполнения.

Таким образом, получена оценка числа процессоров $n = 2$, позволяющая выполнить алгоритм оценки элемента популяции за минимальное время $T = T_{кр}$. Данная оценка является решением задачи, так как в матрице независимости нет множеств ВНО, содержащих число операторов $r > n$ в соответствии с алгоритмом [9].

Приведем некоторые экспериментальные результаты, полученные при реализации ГА криптоанализа, проводимого с использованием процессо-

Таблица 3

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1																			
2																			
3				1	1	1	1	1	1	1									
4			1		1	1	1	1	1	1									
5			1	1		1	1	1	1	1									
6			1	1	1		1	1	1	1									
7			1	1	1	1		1	1	1									
8			1	1	1	1	1		1	1									
9			1	1	1	1	1	1		1									
10			1	1	1	1	1	1	1										
11																			
12																			
13																			
14																			
15																			
16																	1	1	
17																1			1
18																1			1
19																	1	1	

Таблица 4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1											1	1	1	1	1	1	1	1	1
2			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3		1		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4		1	1		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5		1	1	1		1	1	1	1	1	1	1	1	1	1	1	1	1	1
6		1	1	1	1		1	1	1	1	1	1	1	1	1	1	1	1	1
7		1	1	1	1	1		1	1	1	1	1	1	1	1	1	1	1	1
8		1	1	1	1	1	1		1	1	1	1	1	1	1	1	1	1	1
9		1	1	1	1	1	1	1		1	1	1	1	1	1	1	1	1	1
10		1	1	1	1	1	1	1	1		1	1	1	1	1	1	1	1	1
11	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1	1
12	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
13	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	

ра CORE I7-4820K, CPU 3,7 GHz, ОЗУ 64 Гбайт. Результаты для двух серий экспериментов представлены в табл. 5, 6.

При реализации эксперимента задавались следующие параметры: размер начальной популяции – 1000; количество итераций – 100; норма мутации и инверсии – 0.05; тип кроссинговера – простой двухточечный.

В 1-м столбце табл. 5, 6 показан номер итерации, во 2-м – количество хромосом, подвергнувшихся мутации и инверсии, в столбцах 3–12 –

Таблица 5

0	1000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
1	1800	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
2	2592	25.000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
3	3731	25.000	25.000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
4	5372	25.000	25.000	25.000	12.500	12.500	12.500	12.500	12.500	12.500	12.500
5	7735	25.000	25.000	25.000	25.000	25.000	12.500	12.500	12.500	12.500	12.500
6	11 138	25.000	25.000	25.000	25.000	25.000	25.000	25.000	12.500	12.500	12.500
7	16 038	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
8	23 094	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	12.500
9	33 255	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
10	47 888	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
11	68 958	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
12	99 298	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
13	142 988	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
14	205 902	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
15	296 497	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
16	426 955	37.500	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
17	614 816	37.500	37.500	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000
18	885 334	37.500	37.500	37.500	37.500	25.000	25.000	25.000	25.000	25.000	25.000
19	1 274 881	37.500	37.500	37.500	37.500	37.500	37.500	25.000	25.000	25.000	25.000
20	1 835 828	37.500	37.500	37.500	37.500	37.500	37.500	37.500	25.000	25.000	25.000
21	2 643 592	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500
22	3 806 771	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500
23	5 481 748	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500
24	7 893 716	50.000	50.000	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500
32	15 524 316	62.500	50.000	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500

Таблица 6

0	1000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
1	1800	25.000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
2	2592	25.000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
3	3731	25.000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
4	5372	25.000	25.000	12.500	12.500	12.500	12.500	12.500	12.500	12.500	12.500
5	7735	25.000	25.000	25.000	25.000	12.500	12.500	12.500	12.500	12.500	12.500
6	11 138	25.000	25.000	25.000	25.000	25.000	25.000	25.000	12.500	12.500	12.500
7	16 038	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
8	23 094	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
9	33 255	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
10	47 888	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
11	68 958	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
12	99 298	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
13	142 988	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
14	205 902	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
15	296 497	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
16	426 955	37.500	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
17	614 816	37.500	37.500	25.000	25.000	25.000	25.000	25.000	25.000	25.000	25.000
18	885 334	37.500	37.500	37.500	37.500	25.000	25.000	25.000	25.000	25.000	25.000
19	1 274 881	37.500	37.500	37.500	37.500	37.500	25.000	25.000	25.000	25.000	25.000
20	1 835 828	37.500	37.500	37.500	37.500	37.500	25.000	25.000	25.000	25.000	25.000
21	2 643 592	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	25.000	25.000
22	3 806 771	50.000	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500
23	5 481 748	50.000	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500
24	7 893 716	50.000	50.000	37.500	37.500	37.500	37.500	37.500	37.500	37.500	37.500
33	15 956 745	62.500	62.500	50.000	50.000	50.000	50.000	37.500	37.500	37.500	37.500

значение в процентах для 10 лучших хромосом популяции, определяющее совпадение полученного текста с исходным. Как видно из таблиц, на 24-й генерации наилучшая хромосома обеспечивает совпадение полученного текста с исходным на 50 %, на 32–33-й генерациях – на 62.5 %.

Время реализации алгоритма для получения квазиоптимального ключа (50 %) составило при двухточечном кроссинговере (мутации и инверсии 5 %) от 74 мин до 24 ч.

Приведем результаты эксперимента по определению квазиоптимального ключа, обеспечивающего максимальное совпадение полученного текста с исходным. В качестве исходного был использован следующий текст:

«*У лукоморья дуб зеленый, златая цепь на дубе том; и днем и ночью кот ученый все ходит по цепи кругом; идет направо – песнь заводит, налево – сказку говорит*.»

При реализации алгоритма криптоанализа посредством разбиения исходного текста на 8-буквенные блоки и использования параллельного вычислительного процесса был определен квазиоптимальный ключ, обеспечивающий получение следующего текста:

*«У*лук*м**ья*ду* _***еный_*ла*а*_ц*
пь_нд*б*_то*;_и*д*ем***н*чью*
***т_*чены*_в*е*х*д*т_по*цеп**кр**
го;*ид**_н*п*аво_**пе*н*_зав**и*
*т**на*ево***сказ*у_*ов*рит.*_**_»*

Как можно заметить, полученный текст достаточно близок к исходному (совпадение в пределах 62.5 %), содержит почти осмысленные слова (д*ем, н*чью, сказ*у), из чего следует, что процесс расшифровки (например, при использовании ГА для криптоанализа 1-го типа) может быть доведен до конца вручную (аналогично примеру, описанному в [6]).

Таким образом, в данной статье описано применение ГА для реализации криптоанализа блочных криптосистем на примере стандарта шифрования ГОСТ, приведены результаты эксперимента при реализации криптоанализа 2-го типа на основе параллельной схемы его реализации. Как показали результаты эксперимента, полученные результаты по определению оптимального ключа (при криптоанализе 2-го типа) в общем случае в значительной степени зависят от длины исходного текста, что может привести к эффективному использованию вычислительных систем, допускающих параллельную обработку информации (в частности, многопроцессорных систем класса SIMD).

Работа выполнена при финансовой поддержке РФФИ (проекты 14-01-00634, 15-01-05129).

СПИСОК ЛИТЕРАТУРЫ

1. Криптографические методы и генетические алгоритмы решения задач криптоанализа / Ю. О. Чернышев, А. С. Сергеев, Е. О. Дубров, О. П. Третьяков / ФВАС. Краснодар, 2013. 138 с.

2. Авдошин С. М., Савельева А. А. Криптоанализ: современное состояние и перспективы развития // Прил. к журн. «Информационные технологии». 2007. № 3. С. 1–32.

3. Бабенко Л. К., Ищукова Е. А. Современные алгоритмы блочного шифрования и методы их анализа. М.: Гелиос АРВ, 2006. 376 с.

4. Чернышев Ю. О., Сергеев А. С., Дубров Е. О. Обзор алгоритмов решения задач криптоанализа на основе биоинспирированных технологий искусственного интеллекта // Вестн. Воронеж. гос. ун-та. 2014. № 2. Сер. Системный анализ и информационные технологии. С. 83–89.

5. Разработка метода криптоанализа блочных шифров в системах защиты информации на основе параллельного генетического поиска / Ю. О. Чернышев, А. С. Сергеев, А. Н. Рязанов, В. М. Москалев // Сб. докл. XVII Междунар. конф. по мягким вычислениям и измерениям. Т. 1. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2015. С. 408–411.

6. Исследование возможности применения генетических алгоритмов для реализации криптоанализа блочных криптосистем / Ю. О. Чернышев, А. С. Сергеев, Н. Н. Венцов, А. Н. Рязанов // Вестн. Донского гос. техн. ун-та. 2015. № 3(82). С. 65–72.

7. Сергеев А. С. Разработка методов криптоанализа на основе генетического поиска при реализации стратегий и технологий информационной защиты на примере стандарта шифрования России // Коммуникативные стратегии информационного общества: тр. междунар. науч.-техн. конф. СПб.: Изд-во Политехн. ун-та, 2007. С. 56–65.

8. Романец Ю. В., Тимофеев П. А., Шаньгин В. Ф. Защита информации в компьютерных системах и сетях. М.: Радио и связь, 1999. 328 с.

9. Сергеев А. С. Параллельное программирование. Ростов н/Д.: Издательский центр ДГТУ, 2002. 77 с.

10. Ахо Альфред В., Джон Ульман Джеффри Д. Структуры данных и алгоритмы. М.: Издательский дом «Вильямс», 2003. 384 с.

Yu. O. Chernyshev, A. S. Sergeev, A. N. Rjazanov
Don state technical university

S. A. Kapustin
Krasnodar higher military school

RESEARCH OF POSSIBILITY OF APPLICATION OF EVOLUTIONARY OPTIMIZATION METHODS FOR REALIZATION OF CRYPTANALYSIS OF ENCIPHERING BLOCK METHODS

The problem of cryptanalysis of cryptographic protection methods with use of new model of optimizing methods – the genetic algorithms imitating processes of evolution of wildlife is considered. Application of genetic algorithms for cryptanalysis of the Russia block standard of enciphering is described. The block diagram of genetic algorithm, the description and distinctive features of process of cryptanalysis, results of experimental realization are provided.

Cryptanalysis, the bioinspired methods, genetic algorithm, block algorithm of enciphering, population of keys, a crossing over, a quasi-optimal key