УДК 519

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ГЕНЕТИЧЕСКОГО АЛГОРИТМА НА ОСНОВЕ МОДЕЛИ ГОЛДБЕРГА С АНАЛИЗОМ ЕГО ПРИМЕНЕНИЯ В КРИПТОГРАФИИ

Ясашный О. П., Сафарьян О. А.
Донской государственный технический университет (г. Ростов-на-Дону, Российская Федерация)

Проанализировано применение генетических алгоритмов в криптографии. Представлена классификация алгоритмов шифрования, подробно описан алгоритм RSA и способы его криптоанализа. Разработано программное средство для факторизации большого составного числа на два простых множителя с помощью генетического алгоритма на основе модели Голдберга. Результаты вычислений программы представлены в таблице и проанализированы.

***Ключевые слова*:** криптография, криптоанализ, алгоритм RSA, факторизация чисел, генетический алгоритм, модель Голдберга.

SOFTWARE IMPLEMENTATION OF THE GENETIC ALGORITHM BASED ON THE GOLDBERG MODEL WITH THE ANALYSIS OF ITS APPLICATION IN CRYPTOGRAPHY

Oleg P. Yasashnyi, Olga A. Safaryan
Don State Technical University (Rostov-on-Don, Russian Federation)

The paper analyzes the application of genetic algorithms in cryptography. A classification of encryption algorithms is presented. The RSA algorithm and methods of its cryptanalysis are described in detail. A software tool has been developed for factorization of a large composite number into two prime factors using a genetic algorithm based on the Goldberg model. The calculation results of the program are presented in the table and analyzed.

***Keywords*:** cryptography, cryptanalysis, RSA algorithm, number factorization, genetic algorithm, Goldberg model.

Введение. В настоящее время проблема информационной безопасности и защиты обрабатываемой информации становится все более актуальной. Каждый пользователь компьютера стремится обезопасить свои личные данные [1].

Проблемой обеспечения конфиденциальности, целостности и доступности данных занимается криптография. Для защиты информации от посторонних лиц обычно применяется шифрование, которое преобразовывает данные в нечитаемый набор символов по определённому алгоритму. В зависимости от количества используемых ключей, алгоритмы шифрования делятся на 2 типа:

– система с закрытым ключом (симметричная). Для шифрования и расшифрования текста используется один закрытый ключ. Его необходимо держать в секрете и передавать только через защищенные каналы связи, так как при перехвате или потере ключа злоумышленник сможет получить доступ к секретной информации;

– система с открытым ключом (асимметричная). В ней используется пара ключей, которая представляет собой связку из открытого и закрытого ключа. Первый из них необходим для шифрования исходного сообщения. Он передаётся по открытым каналам связи и может являться общедоступным. Закрытый ключ необходим для расшифровки шифротекста, его необходимо держать в секрете. К этой системе также можно отнести алгоритм электронной подписи (ЭП). Для ее генерации используется закрытый ключ, а для проверки ЭП — открытый.

Сейчас наиболее актуальными являются асимметричные системы, они применяются в таких криптографических протоколах, как TLS, S/MIME, PGP. Одной из первых ассиметричных криптосистем является RSA. Алгоритм был придуман в 1977 году учеными Ривестом, Шамиром и Адлеманом [2]. Несмотря на то, что RSA был изобретен 45 лет назад, он до сих пор остается актуальным. Криптосистема основывается на вычислительной сложности факторизации больших целых чисел, что относится к классу NP задач [2].

В современном мире для решения NP и NP-полных задач все более актуальными становятся алгоритмы, основанные на природных системах, к ним относятся: генетические, пчелиные, муравьиные алгоритмы, а также метод роя частиц.

Генетический алгоритм — это эвристический алгоритм поиска, используемый для решения задач оптимизации и моделирования путем случайного подбора, комбинирования и вариации искомых параметров с использованием механизмов, аналогичных естественному отбору в природе [3].

Целью статьи является программная реализация генетического алгоритма на основе модели Голдберга с анализом его применения в криптографии.

Основная часть. RSA может использоваться как для процессов шифрования и расшифрования, так и для электронной подписи. Процесс генерации ключей одинаковый для обоих алгоритмов и состоит в следующем:

1. Генерируются два больших простых числа p и q, они не должны быть равны друг другу.
2. Число n находится как произведение чисел p и q.
3. От числа n находится функция Эйлера по следующей формуле:

$$\varphi(n) = (q-1)(p-1). \quad (1)$$

4. Число e называется открытой экспонентой. Ее значением является случайное целое число, для которого должны выполняться два условия: оно должно быть взаимно простым с $\varphi(n)$, $1 < e < \varphi(n)$;
5. Число d является закрытой экспонентой и вычисляется по формуле:

$$ed = 1 \left(mod\ \varphi(n)\right). \quad (2)$$

Это число можно найти, используя, например, расширенный алгоритм Евклида.

Получившаяся пара чисел e и n используется как открытый ключ, а пара d и n — как закрытый.

Для того, чтобы зашифровать сообщение m, необходимо его десятичное представление возвести в степень открытой экспоненты:

$$c = m^e\ (mod\ n). \quad (3)$$

Для обратного преобразования шифртекст надо возвести в степень d:

$$m = c^d\ (mod\ n). \quad (4)$$

Для криптоанализа RSA, т.е. получения исходного текста без знания закрытого ключа, возможно несколько вариантов:

1. Если известен открытый текст и его шифр, то можно подобрать такую закрытую экспоненту, чтобы выполнялось равенство (4).
2. Подобрать значение функции Эйлера от числа n, а затем найти d из выражения (2) [4].
3. Подобрать простые большие числа p и q, чтобы их произведение равнялось модулю n.

Третий вариант имеет наименьшую комбинаторную сложность. Алгоритмы факторизации являются субэкспоненциальными и экспоненциальными, поэтому для поиска простых делителей в статье используются генетические алгоритмы (ГА), которые способны дать точный результат за приемлемое время.

Одним из ключевых элементов ГА является особь (индивид), которая представляет одно из возможных решений данной задачи. Необходимо выбрать корректную структуру, так как от неё зависит эффективность работы алгоритма. Индивид обычно представлен в виде битовой строки, каждый элемент которой называется геном и принимает значение 0 и 1. Совокупность генов образует хромосому. Для разложения числа на два простых сомножителя возможны несколько вариантов представления особи:

– индивид представлен двумя хромосомами, первая из которых представляет число p в двоичном виде, а вторая — число q;

– индивид представлен одной хромосомой, которая представляет один из множителей. Так как разложение числа n единственно, то зная первый сомножитель, найти второй можно из уравнения

$$q = \frac{n}{p}. \quad (5)$$

Второе представление является более эффективным и простым, а также занимает меньше памяти.

Для оценки решения, представляемого особью, создана функция приспособленности. Она является аналогом меры пригодности индивида к выживанию в естественной среде обитания. Чем меньше её значение, тем лучше особь. Для представления с одной хромосомой значение рассчитывается по следующей формуле:

$$fitness\ function(i) = n\ (mod\ \ i), \quad (6)$$

где i — десятичное представление генов особи.

Если значение функции будет равно 0, то мы нашли один из делителей n. Главном недостатком данной функции является проблема нахождения экстремума немонотонной функции.

Алгоритм модели Голдберга:

– инициализация начальной популяции;

– выбор родителей для скрещивания;

– получение потомком с помощью кроссовера (скрещивания) родителей. Далее происходит мутация потомков;

– обновление популяции. Поиск лучшей особи по значению функции приспособленности. Если её приспособленность повторяется заданное количество раз, происходит конец алгоритма, иначе повторить шаги 2–4.

Создание начальной популяции заключается в формировании заданного количества особей, гены которых генерируются случайным образом. Для более эффективной работы алгоритма для каждого индивида необходимо сгенерировать большое простое число заданного размера, так как по условию задачи оба множителя являются простыми. Простое число генерировалось на основе вероятностного теста простоты Рабина-Миллера.

В данной статье использовался двухточечный кроссовер. Алгоритм его работы:

– выбор двух границ случайным образом;

– гены первого родителя до первой и после второй границы переходят в гены потомка;

– гены второго родителя в промежутке между границами переходят в гены потомка. Схема двухточечного кроссовера представлена на рис. 1.

В качестве мутации был выбран метод перетасовки. Он заключается в изменении порядка некоторой части генов на случайный. Длина изменяемого участка составляет 10 процентов от общей длины генов.

После выполнения операторов кроссовера и мутации может получиться индивид, гены которого будут представлять непростое число. Это решение является невалидным, поэтому, после

выполнения классических генетических операций, необходимо осуществить поиск ближайшего простого числа к текущему решению в интервале [p – r, p + r], где p — это десятичное представление особи, а r — радиус поиска, который вычисляется по формуле [5]:

$$r = \frac{m}{1{,}442695}, \quad (7)$$

где m — длина числа n в битах.

Проверка числа на простоту выполняется с помощью теста Рабина-Миллера [6].

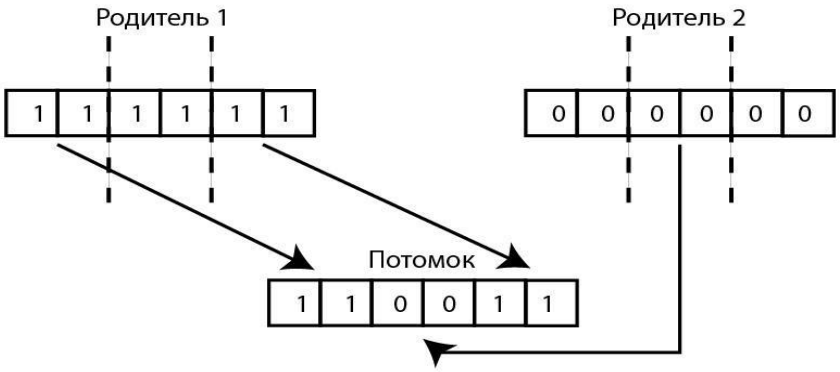

Рис. 1. Двухточечный кроссовер

В таблице 1 представлены результаты разложения большого числа n на два простых множителя p и q с помощью генетического алгоритма на основе модели Голдберга. Для повышения эффективности работы алгоритма размер поколений увеличивался пропорционально длины числа n в битах. Если при повторении лучшего решения 1000 раз сомножитель не был найден, то запуск считается неуспешным.

Таблица 1

Результаты вычислительного эксперимента

n	Длина n в битах	p	q	Размер поколений	Количество успешных запусков
525671040119	40	653143	804833	500	25/25
8683975841033	44	2286149	3798517	500	25/25
140430962809127	48	15195737	9241471	1000	25/25
3165200596700783	52	66792641	47388463	5000	25/25
25108261821576161	56	151634059	165584579	10000	25/25
657219588816551759	60	987423677	665590267	10000	25/25
1917326574694467641	62	1733747903	1105885447	10000	25/25
9911278386797819761	64	2519866603	3933255187	15000	25/25
53874900861547998143	66	7570262983	7116648521	15000	20/25
152652515933753447329	68	11177861231	13656683759	20000	10/25
452654067668499678431	70	21945171553	20626590527	20000	3/25

Заключение. В ходе проделанной работы было проанализировано применение генетических алгоритмов в криптографии. Разработано программное средство криптоанализа алгоритма RSA на основании факторизации большого составного числа на два простых множителя с помощью генетического алгоритма на основе модели Голдберга. Наилучшим результатом работы программы является успешное разложение большого числа, длина которого

составила 70 бит, на 2 простых множителя. В результате можно отметить, что данная программная реализация применима для криптоанализа системы RSA и факторизации большого составного числа на два простых множителя.

Библиографический список

1. Сергеев, А. С. О возможности применения методов генетического поиска для реализации криптоанализа асимметричного алгоритма шифрования данных RSA / А. С. Сергеев // Известия высших учебных заведений. Северо-Кавказский регион. Технические науки. — 2008. — № 3(145). — С. 48–52.

2. Омассон, Ж. О криптографии всерьез / Ж. Омассон. — Москва : ДМК Пресс, 2022. — 328 с.

3. Каширина, И. Л. Введение в эволюционное моделирование / И. Л. Каширина. — Воронеж : Воронежский государственный университет, 2007. — 40 с.

4. Биоинспирированные методы криптоанализа асимметричных алгоритмов шифрования на основе факторизации составных чисел / А. С. Сергеев, О. П. Третьяков, А. Е. Васильев, Ю. О. Чернышев // Вестник Донского государственного технического университета. — 2011. — Т. 11. — № 9(60). — С. 1544–1554.

5. Кажаров, Х. А. Разработка генетической модели поиска простых чисел для криптоанализа RSA на основе клиент-серверной структуры / Х. А. Кажаров // Известия ЮФУ. Технические науки. — 2008. — № 9(86). — С. 40–46.

6. Шнайер, Б. Прикладная криптография. Протоколы, алгоритмы и исходный код на С / Б. Шнайер — Москва : Диалектика, 2016. — 1024 с.

Об авторе:

Ясашный Олег Петрович, студент кафедры «Компьютерная безопасность» Донского государственного технического университета (344003, РФ, г. Ростов-на-Дону, пл. Гагарина, 1), oleg999000@yandex.ru

Сафарян Ольга Александровна, доцент кафедры «Компьютерная безопасность» Донского государственного технического университета (344003, РФ, г. Ростов-на-Дону, пл. Гагарина, 1), кандидат технических наук, доцент, safari_2006@mail.ru

About the Authors:

Yasashnyi, Oleg P., Student, Computer Security Department, Don State Technical University (1, Gagarin sq., Rostov-on-Don, 344003, RF), oleg999000@yandex.ru

Safaryan, Olga A., Associate Professor, Computer Security Department Don State Technical University (1, Gagarin sq., Rostov-on-Don, 344000, RF), Cand.Sci. (Eng.), Associate Professor, safari_2006@mail.ru