Инженерный вестник Дона, №4 (2024)
ivdon.ru/ru/magazine/archive/n4y2024/9139

Искусственные иммунные системы в криптоанализе и решении диофантовых уравнений: новый подход к защите информации

В.И. Шиян, Ю.А. Аносова, Н.О. Дворников, М.Е. Кокоровец, Ю.Д. Кривошей, Д.С. Самусев, М.С. Щербина

Кубанский государственный университет, Краснодар

Аннотация: В статье рассматривается задача криптоанализа системы защиты информации, основанной на трудно решаемой задаче диофантовых уравнений. Описывается математическая модель такой системы защиты и предлагается решение задачи криптоанализа с использованием искусственной иммунной системы, адаптированной для решения диофантовых уравнений. В работе рассматриваются основные принципы построения искусственных иммунных систем и приводятся результаты экспериментов по оценке эффективности предложенной системы на диофантовых уравнениях степени, не превышающей шести. Полученные результаты демонстрируют возможность применения искусственных иммунных систем для решения задачи криптоанализа систем защиты информации, основанных на диофантовых уравнениях.

Ключевые слова: криптоанализ, система защиты информации, диофантовы уравнения, искусственная иммунная система, адаптивный алгоритм, оценка эффективности.

Введение

Искусственные иммунные системы (ИИС) в области информатики – это адаптивные алгоритмы, которые могут использовать принципы иммунологии для решения различных задач. Они опираются на теоретические основы негативного отбора, клональной селекции и работы иммунной системы, которые лежат в основе алгоритмов функционирования этих систем. Использование ИИС широко применяется в различных областях, в том числе в криптоанализе систем защиты информации (СЗИ).

Криптоанализ, являющийся разделом криптологии, фокусируется на изучении методов декодирования шифров и незаконного доступа к информации [1, 2].

СЗИ обеспечивают надёжную защиту данных на всех этапах их обработки и передачи.

Одним из подходов к построению СЗИ является использование диофантовых трудностей (ДТ), которые основаны на сложности решения диофантовых уравнений (ДУ). ДУ представляет собой уравнение вида (1):

$$D\left(x_1,\, x_2,\, \ldots,\, x_m\right)=0, \tag{1}$$

где D – полиномиальная функция с целыми коэффициентами, и x_i являются переменными, принимающими значения из множества целых чисел [3-5].

СЗИ, содержащие ДТ, используют сложность решения ДУ в качестве основания для создания криптографических алгоритмов [6].

Формулировка задачи определения решений ДУ

В процессе анализа разрешимости ДУ переменные классифицируются на параметры (которые предполагаются постоянными) и неизвестные. Уравнение вида:

$$D\left(a_1,\, a_2,\, \ldots,\, a_n,\, x_1,\, x_2,\, \ldots,\, x_m\right)=0, \tag{2}$$

где $a_1,\, a_2,\, \ldots,\, a_n$ являются параметрами, а $x_1,\, x_2,\, \ldots,\, x_m$ – неизвестными, считается разрешимым при определённых условиях для параметров $\left(a_1,\, a_2,\, \ldots,\, a_n\right)$, если существует такой набор значений неизвестных $\left(x_1,\, x_2,\, \ldots,\, x_n\right)$, который обращает уравнение в верное равенство [7, 8].

Применение ИИС для решения ДУ

В данной работе рассматриваются ключевые аспекты процедур иммунной оптимизации.

1) Процедуры, применяемые в рамках иммунной оптимизации, осуществляются на уровне популяции клеток:

$$P=\left\{p_i,\, i\in\left[1;\, n_p\right]\right\}, \tag{3}$$

где каждая отдельная клетка $p_i=\left(p_{i,\,1},\, p_{i,\,2},\, \ldots,\, p_{i,\,m}\right)$ представляет собой совокупность различных компонентов $p_{i,\,1},\, p_{i,\,2},\, \ldots,\, p_{i,\,m}$, которые являются

случайными целыми числами из отрезка $[-10; 10]$, причём количество этих компонентов соответствует числу неизвестных в ДУ. Каждая переменная кодируется одной компонентой. Общее количество сгенерированных клеток составляет n_p, при этом параметр n_p определяется пользователем.

2) В популяции реализуются процедуры клонирования и мутации клеток, при этом количество процедур мутации может достигать трёх. Механизм клонирования обеспечивает формирование копий, полностью идентичных исходной клетке:

$$C_i = \left\{ p_j^{C_i} : p_j^{C_i} = p_i,\ j \in [1;\ n_C] \right\}, \tag{4}$$

где $p_j^{C_i}$ – j-й клон клетки p_i, и $n_C = 10$ – число клонов, образованных каждой клеткой в популяции P.

Целевая функция описывается следующим образом: $f(x_1, x_2, \ldots, x_m) = D(x_1, x_2, \ldots, x_m)$, где $f(p)$ – значение функции, соответствующее определённой клетке p, и $D(x_1, x_2, \ldots, x_m) = 0$ представляет собой ДУ, требующее решения.

При мутации компоненты вектора $p_i^{C_i}$ изменяются случайным образом:

$$p_{j,k}^{C_i} = p_{j,k}^{C_i} + \Delta_{j,k},\ k \in [1;\ n], \tag{5}$$

где $\Delta_{j,k}$ – шаг мутации. В случае, когда значение целевой функции для определённой клетки не достигает нулевого значения, инициируется серия процедур мутации. Первая процедура в данной серии предполагает модификацию значения выбранной наугад переменной путём её замены на произвольно сгенерированное целочисленное значение из отрезка $[-1000; 1000]$. В ситуации, когда после выполнения первой процедуры мутации значение целевой функции сохраняет ненулевое значение, активируется последующая процедура мутационных манипуляций. Данная стадия предусматривает корректировку значения выбранной наугад

переменной путём её замены на произвольно сгенерированное целочисленное значение из отрезка $\left[\left[med\right];\left[mn\right]\right]$, если $\left[med\right]<\left[mn\right]$, и отрезка $\left[\left[mn\right];\left[med\right]\right]$ в противном случае, где med – медиана и mn – среднее арифметическое чисел, задействованных в ДУ, а $\left[x\right]$ – целая часть заданного числа x. В случае сохранения ненулевого значения функционала оптимизации даже после второй процедуры мутации, применяется третья процедура мутационного процесса. Этап включает в себя корректировку значения выбранной наугад переменной путём её замены на произвольно сгенерированное целочисленное значение из отрезка $\left[-5;\ 5\right]$.

3) Во множество клеток памяти M добавляются клетки, у которых значение целевой функции равно нулю.

4) В результате процедуры клональной селекции наилучшие потомки замещают родителей в соответствии с формулой

$$P=\left\{p_i\in P\ \middle|\ \exists S\subseteq P,\ \left|S\right|=n_p,\ p_i\in S,\ \forall p_j\in S,\ \forall p_k\in P\setminus S,\ f\left(p_j\right)\le f\left(p_k\right)\right\}. \quad (6)$$

5) Процедура сжатия популяции заключается в удалении лишних элементов путём отбора и исключения наихудшего из решений p_j, p_k, при определённых условиях:

$$\left\|p_j-p_k\right\|<b_r,\ j\ne k, \quad (7)$$

где $\left\|\bullet\right\|$ – символ меры близости клеток, евклидова норма, и $b_r=0,2$ – порог сжатия.

6) Генерируется всего max_pop популяций. Параметр max_pop задаётся пользователем.

7) Пока не сгенерировано достаточное количество популяций, то переходим к шагу 2.

Клонирование, мутация и сжатие популяции помогают улучшить оптимизацию и сократить вычислительные расходы.

Для изучения функционирования ИИС была разработана программа на языке Python, предназначенная для выполнения ранее описанных процедур [9]. Результат работы данной программы демонстрируется на примере решения ДУ $x^3 + y^3 + z^3 = 2$, и представлен на рис. 1.

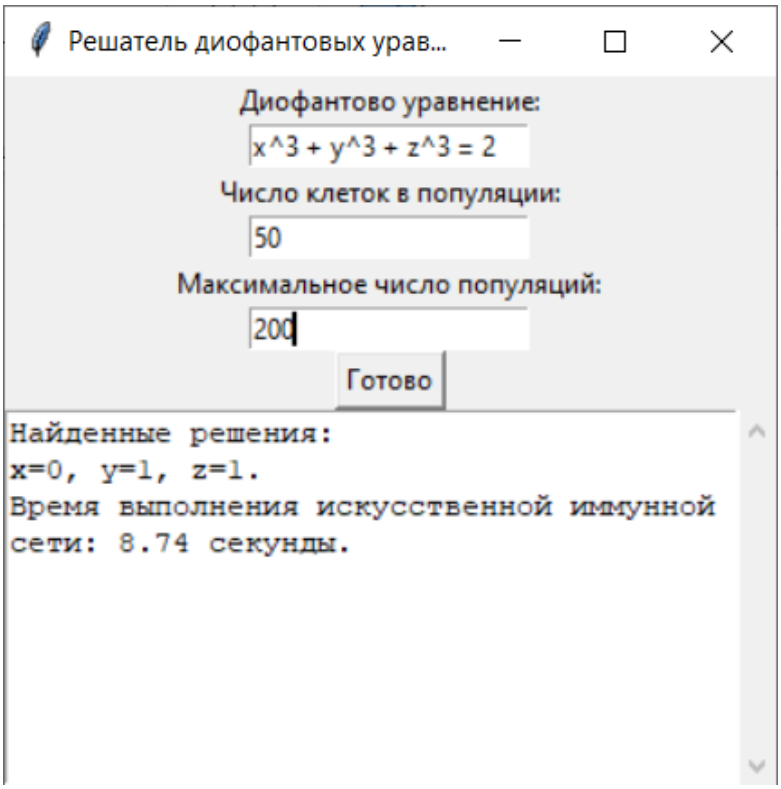

Рис. 1. – Результат функционирования ИИС

В последующих анализах использовались ДУ различных степеней, включая линейное – $32x + 45y = 779$, квадратное – $x^2 + y^2 = z(x + y)$, кубическое – $x^3 = y^2 + z$, четвёртой степени – $9x^4 + 7y = 16$, пятой степени – $x^5 + 13y = 14$ и шестой степени – $x^6 + y = 65$.

Был выполнен анализ качества работы ИИС от параметра n_p (количества клеток в популяции). На рис. 2 представлена зависимость качества работы ИИС от количества клеток в популяции при $max\_pop = 200$.

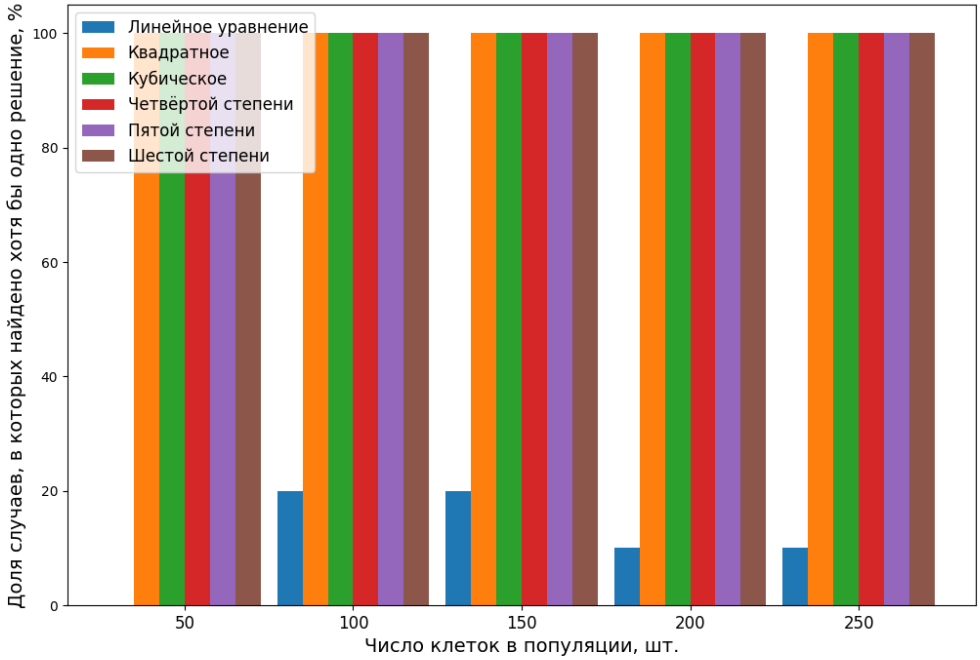

Рис. 2. – Зависимость качества работы ИИС от количества клеток в популяции

Данные рис. 2 демонстрируют, что, за исключением линейного ДУ, все остальные типы ДУ в среднем решаются в 100% случаев при $n_p = 50,\ 100,\ 150,\ 200,\ 250$. Это может объясняться тем, что линейное ДУ содержит большие по модулю числа. К примеру, решение линейного ДУ $10x + 7y = 19$ возможно в 100% случаев при $n_p = 50,\ 100,\ 150,\ 200,\ 250$. Параметр n_p изменяется с шагом 50, что является адекватным для проведения исследования. Уменьшение n_p приводит к уменьшению времени работы алгоритма и количества перебираемых клеток. Оптимальное значение $n_p = 100$ – это минимальное значение, при котором различные ДУ находят своё решение в среднем во всех случаях.

Было установлено, что оптимальное значение параметра max_pop (максимальное число популяций) составило 150. Все ДУ, за исключением

линейного ДУ, решались в среднем в 100% случаев при $max\_pop = 50,\ 100,\ 150,\ 200,\ 250$. Изменение параметра max_pop происходило с шагом в 50 единиц, что оказалось вполне подходящим для данного исследования. Время выполнения алгоритма прямо пропорционально количеству популяций, и 150 оказалось минимальным значением из всех исследованных, при котором все ДУ в среднем решались во всех случаях. Полученные данные свидетельствуют о наличии квадратичной зависимости между временной сложностью алгоритма и общим числом клеток в популяции.

Математическая модель (ММ) СЗИ, содержащей на ДТ

ММ СЗИ, содержащей ДТ представляет собой набор (8)

$$X_0 = \left\langle S^*,\ N,\ T^*,\ A(s),\ B(t) \mid Z\left(A(s),\ B(t)\right)\right\rangle, \tag{8}$$

где S^* – множество цепочек символов $s = s_1 s_2 \ldots s_l$ из алфавита S, представляющее возможные сообщения, где S – множество символов, которые могут использоваться в сообщениях, эти символы могут быть как числовыми, так и буквенными; N – множество числовых кодов (ЧК) для элементарных сообщений (ЭС) s_i из S^*, ЭС может быть как символом, так и цепочкой символов из алфавита S; T^* – множество цепочек символов $t = t_1 t_2 \ldots t_l$ из алфавита T, полученных через прямое преобразование (ПП) цепочки s в t; $A(s)$ – алгоритм ПП цепочки символов $s \in S^*$ в $t \in T^*$; $B(t)$ – алгоритм обратного преобразования (ОП), позволяющий восстановить $s \in S^*$ из $t \in T^*$; $Z\left(A(s),\ B(t)\right)$ – соотношение между алгоритмами $A(s)$ и $B(t)$ таково, что для каждой цепочки $s = s_1 s_2 \ldots s_l \in S^*$ единственным образом получается соответствующая цепочка $t = t_1 t_2 \ldots t_l \in T^*$, и наоборот.

В [10, 11] предлагается ММ СЗИ, содержащей ДТ. Данная СЗИ строится на основе задачи нахождения корней ДУ. Алгоритмы ПП $A(s)$ и ОП $B(t)$ строятся с учётом корней данного ДУ. В рамках данного исследования в качестве примера рассматривается ДУ (9)

$$x^3 + y^3 + z^3 = 1, \tag{9}$$

для которого одно из семейств решений представлено в формуле (10)

$$x = 1 - 9b^3,\ y = 3b - 9b^4,\ z = 9b^4, \tag{10}$$

где b является произвольным целым числом.

Доказательство корректности формулы (10) следует из формулы (11)

$$\begin{gathered}\left(1-9b^3\right)^3 + \left(3b-9b^4\right)^3 + \left(9b^4\right)^3 = 1 - 27b^3 + 243b^6 - 729b^9 + \\ 27b^3 - 243b^6 + 729b^9 - 729b^{12} + 729b^{12} = 1.\end{gathered} \tag{11}$$

Из формул (4) и (5) следует, что формула (12)

$$\left(1-9b^3\right)^3 + \left(3b-9b^4\right)^3 = 1 - \left(9b^4\right)^3, \tag{12}$$

справедлива для произвольного целого числа b.

Обозначим через $M = \{A, B, \ldots, Z, \_\}$ алфавит, а $Q = \{1, 2, \ldots, 27\}$ является набором ЧК букв алфавита M. Определим цепочку символов $m = \text{IT\_IS\_A\_SECRET}$. В этом контексте вводятся функции ПП ЭС (13) и ОП ЭС (14):

$$C_L(b) = \left(1-9b^3\right)^3 + \left(3b-9b^4\right)^3, \tag{13}$$

где b – элемент множества Q,

$$C_R(b) = 1 - \left(9b^4\right)^3, \tag{14}$$

где b – элемент множества Q.

Процесс дешифрования первой буквы в цепочке символов m (буквы I), ЧК которой равен девяти, осуществляется с использованием $C_L(9) = -205891132094648$. Аналогичный подход применяется для

дешифрования всех последующих букв. Для того чтобы легальный пользователь мог определить первую букву, ему необходимо решить простое ДУ (15), в то время как нелегальному пользователю необходимо решить более сложное ДУ (16)

$$1-\left(9b^4\right)^3=-205891132094648, \tag{15}$$

$$\left(1-9b^3\right)^3+\left(3b-9b^4\right)^3=-205891132094648. \tag{16}$$

Криптоанализ с применением ИИС

Дешифруем цепочку m, начиная с первой буквы (буквы I), применяя ИИС. Для этого ДУ (16) преобразуем в систему ДУ (17), поскольку ИИС обладает способностью решать уравнения или системы уравнений, содержащие не менее двух неизвестных:

$$\begin{cases} a=b, \\ \left(1-9a^3\right)^3+\left(3b-9b^4\right)^3+205891132094648=0. \end{cases} \tag{17}$$

Для дешифрования первой буквы необходимо применить ИИС для решения системы ДУ (17). На рис. 3 представлен результат решения ИИС системы ДУ (17).

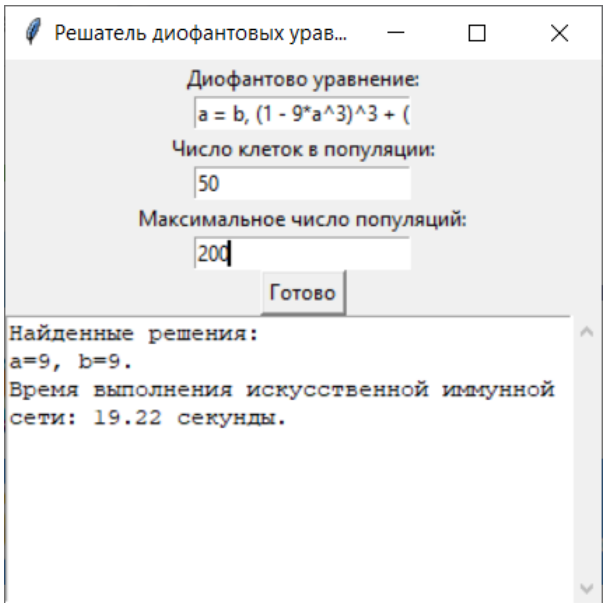

Рис. 3. – Результат решения ИИС системы ДУ (17)

Согласно рис. 3, ИИС нашла решение $a = b = 9$, что подтверждает корректность вычислений, утверждая, что ЧК первой буквы равен девяти, что является правильным результатом. Подобным образом возможно продолжить процесс дешифровки остальных символов в сообщении.

Заключение

В данной работе была разработана ИИС для решения ДУ, возникающих в процессе криптоанализа СЗИ, содержащих ДТ. Созданная ИИС продемонстрировала высокое качество работы и низкую временную сложность при решении ДУ степени не выше шести. Полученные результаты свидетельствуют о перспективности применения ИИС для криптоанализа и решения различных ДУ. Разработанный метод может быть использован для повышения эффективности и скорости анализа СЗИ, а также для решения широкого спектра математических задач, связанных с ДУ.

Литература

1. Саломаа А. Криптография с открытым ключом. – Москва: Мир, 1995. – 318 с.

2. Шнайер Б. Прикладная криптография: Протоколы, алгоритмы, исходные тексты на языке Си: пер. с англ. – Москва: Триумф, 2002. – 816 с.

3. Матиясевич Ю. В. Десятая проблема Гильберта. – М.: Издательская фирма “Физико-математическая литература” ВО Наука, 1993.– 224 с.

4. Серпинский В. О решении уравнений в целых числах / пер. с польск. К. Г. Мельникова. – Москва: Физматлит, 1961. – 88 с.

5. Cassels J. W. S. On a Diophantine Equation // Acta Arithmetica. – 1960. – Vol. 6. – pp. 47–52.

6. Осипян В. О., Спирина С. Г., Арутюнян А. С., Подколзин В. В. Моделирование ранцевых криптосистем, содержащих диофантовую трудность // Чебышевский сборник. – 2010. Т. 11, вып. 1. – С. 209–217.

7. Koblitz N. A Course in Number Theory and Cryptography. – New York: Springer-Verlag, 1987. – 235 p.

8. Lenstra A. K., Lenstra H. W., Lovasz L. Factoring polynomials with rational coefficients // Mathematische annalen. – 1982. – Vol. 261. – pp. 515–534.

9. GitHub - Nikifx/calculator-of-diophantine-equations. URL: github.com/Nikifx/calculator-of-diophantine-equations (дата обращения: 30 марта 2024).

10. Осипян В. О. Разработка математической модели дисимметричной биграммной криптосистемы на основе параметрического решения многостепенной системы диофантовых уравнений // Инженерный вестник Дона, 2020, № 6. URL: ivdon.ru/ru/magazine/archive/N6y2020/6534.

11. Осипян В. О., Литвинов К. И., Жук А. С., Синица С. Г., Багдасарян Р. Х. Алгоритм разработки математической модели дисимметричной биграммной криптосистемы, содержащих диофантовы трудности // Инженерный вестник Дона, 2021, № 9. URL: ivdon.ru/ru/magazine/archive/n9y2021/7202.

References

1. Salomaa A. Kriptografiya s otkrytym klyuchom [Cryptography with a public key]. Moskva, Mir Publ., 1995, 318 p.

2. Schneier B. Prikladnaya kriptografiya: Protokoly, algoritmy, iskhodnye teksty na yazyke Si [Applied cryptography: Protocols, algorithms, source texts in C]. Moskva, Triumph Publ., 2002, 816 p.

3. Matiyasevich, Yu. B. Matiyasevich Yu. B. Desyataya problema Gil'berta. [Hilbert's tenth problem]. Izdatel'skaya firma “Fiziko-matematicheskaya literature”. VO Nauka. 1993. 224 p.

4. Serpinsky W. O reshenii uravneniy v tselykh chislakh [On solving equations in integers]. Moskva, Fizmatlit Publ., 1961, 88 p.

5. Cassels J. W. S. Acta Arithmetica. 1960. Vol. 6. pp. 47–52.

6. Osipyan V. O., Spirina S. G., Arutyunyan A. S., Podkolzin V. V. Chebyshevskiy sbornik, 2010, vol. 11, № 1, pp. 209–216.

7. Koblitz N. A Course in Number Theory and Cryptography. New York: Springer-Verlag, 1987. 235 p.

8. Lenstra A. K., Lenstra H. W., Lovasz L. Mathematische annalen. 1982. Vol. 261. pp. 515–534.

9. GitHub - Nikifx/calculator-of-diophantine-equations. URL: github.com/Nikifx/calculator-of-diophantine-equations (accessed: 03/30/2024).

10. Osipjan V. O., Inzhenernyj vestnik Dona, 2020, № 6. URL: ivdon.ru/ru/magazine/archive/N6y2020/6534.

11. Osipjan V. O., Litvinov K. I., Zhuk A. S., Sinitsa S. G., Bagdasaryan R. Kh. Inzhenernyj vestnik Dona, 2021, № 9. URL: ivdon.ru/ru/magazine/archive/n9y2021/7202.

Дата поступления: 29.02.2024

Дата публикации : 8.04.2024