МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

УДК 510.52

О ГЕНЕРИЧЕСКОЙ СЛОЖНОСТИ ПРОБЛЕМЫ ИЗВЛЕЧЕНИЯ КОРНЯ В ГРУППАХ ВЫЧЕТОВ[1]

А. Н. Рыбалов

Омский государственный университет им. Ф. М. Достоевского, г. Омск, Россия,
Институт математики им. С. Л. Соболева СО РАН, г. Новосибирск, Россия

Генерический подход к алгоритмическим проблемам предложен А. Мясниковым, И. Каповичем, П. Шуппом и В. Шпильрайном в 2003 г. В рамках этого подхода рассматривается поведение алгоритмов на множествах почти всех входов. В данной работе изучается генерическая сложность классической проблемы извлечения корня в группах вычетов $\mathbb{Z}/(m)$, где $m = pq$ — произведение двух простых чисел. Доказывается, что её естественная подпроблема генерически трудноразрешима (то есть трудна для почти всех входов) при условии, что проблема извлечения корня трудноразрешима в классическом смысле.

Ключевые слова: *генерическая сложность, проблема извлечения корня в группах вычетов, вероятностный алгоритм.*

DOI 10.17223/20710410/38/7

ON GENERIC COMPLEXITY OF THE PROBLEM OF FINDING ROOTS IN GROUPS OF RESIDUES

A. N. Rybalov

Omsk State University, Omsk, Russia,
Sobolev Institute of Mathematics, Novosibirsk, Russia

E-mail: alexander.rybalov@gmail.com

Every algorithmic problem used in modern cryptography must satisfy important conditions. First of all, the problem should be easily decidable for legal users and hard for cryptanalysis. In addition, there should be an effective algorithm for generation of hard inputs. In practically used cryptosystems with open key, such inputs are randomly generated on a sufficiently large set of inputs. A problem may be hard only for a small part of the inputs (for example, only for polynomial number of words among exponential number of all binary words). So, the problem is easy for almost all inputs. This observation leads to the concept of generic complexity and computability. In the framework of this approach, the algorithmic problem is considered on some subset of "almost all" inputs. Such inputs form the so-called generic set. The concept of "almost all" can be formalized by the introduction of a natural measure on the set of inputs. The problem can be hard (moreover, algorithmically undecidable) on

[1]Работа поддержана грантом РФФИ № 15-41-04312.

the whole set of inputs, but decidable (moreover, effectively decidable) for the "almost all" inputs. But cryptographic problems must remain hard in the generic case. In this paper, we study the generic complexity of the classical algorithmic problem of cryptography — the problem of extracting a root in the residue groups $\mathbb{Z}/(m)$, where $m = pq$ is the product of two different prime numbers. It is still unknown whether there exists a polynomial algorithm, deciding this problem for all inputs. Moreover, the famous cryptosystem RSA is based on the assumption of its hardness. We prove that this problem is generically undecidable in polynomial time, provided there is no polynomial probabilistic algorithm for its solution in the worst case. There is a plausible hypothesis (P = BPP) that any polynomial probabilistic algorithm can be efficiently derandomized, i.e. that a polynomial deterministic algorithm can be build to solve the same problem.

Keywords: *generic complexity, problem of finding roots in groups of residues, probabilistic algorithm.*

Введение

Алгоритмические проблемы, используемые в криптографии с открытым ключом, обладают рядом жёстких свойств. Прежде всего это свойства, связанные с вычислительной сложностью (проблема должна быть легкоразрешимой для легальных пользователей и трудноразрешимой для криптоанализа). Кроме того, важное значение имеет задача легкой генерации входов, на которых проблема является вычислительно сложной. В практически используемых криптосистемах с открытым ключом такие входы генерируются случайным образом на достаточно большом множестве входов проблемы. Если рассматриваемая проблема окажется труднорешаемой лишь для небольшой части входов (например, только для $O(n^3)$ слов среди всех 2^n бинарных слов), то почти на каждом сгенерированном входе проблема будет легко решаться. Это наблюдение привело к понятию генерической сложности и вычислимости [1]. В рамках этого подхода алгоритмическая проблема рассматривается не на всём множестве входов, а на некотором подмножестве «почти всех» входов. Такие входы образуют так называемое генерическое множество. Понятие «почти все» формализуется введением естественной меры на множестве входных данных. Проблема может быть труднорешаемой (более того, алгоритмически неразрешимой) на всём множестве входов, но разрешимой (более того, легкоразрешимой) на «почти всём» множестве входных данных. В [1, 2] доказано, что таким поведением обладают многие алгоритмические проблемы алгебры, а в [3] построено генерическое множество, на котором разрешима классическая проблема остановки для машин Тьюринга с лентой, бесконечной в одном направлении. Для многих классических NP-полных проблем существуют полиномиальные генерические алгоритмы [4].

С точки зрения современной криптографии интересны такие алгоритмические проблемы, которые, являясь (гипотетически) трудными в классическом смысле, остаются трудными и в генерическом смысле, т. е. для почти всех входов. Это объясняется тем, что при случайной генерации ключей в криптографическом алгоритме происходит генерация входа некоторой трудной алгоритмической проблемы, лежащей в основе алгоритма. Если проблема будет генерически легкоразрешимой, то для почти всех таких входов её можно будет быстро решить и ключи почти всегда будут нестойкими. В [5, 6] проведён генерический анализ двух классических алгоритмических проблем криптографии: проблемы распознавания квадратичных вычетов и проблемы дискретного логарифма.

В данной работе изучается генерическая сложность ещё одной классической алгоритмической проблемы криптографии — извлечения корня в группах вычетов $\mathbb{Z}/(m)$, где $m = pq$ — произведение двух различных простых чисел. До сих пор не известно полиномиальных алгоритмов её решения. Более того, на предположении об её трудноразрешимости основана знаменитая криптосистема RSA с открытым ключом [7, 8]. В данной работе доказывается, что эта проблема генерически неразрешима за полиномиальное время при условии отсутствия полиномиального вероятностного алгоритма для её решения в худшем случае. Существует правдоподобная гипотеза о том, что любой полиномиальный вероятностный алгоритм можно эффективно дерандомизировать, т. е. построить полиномиальный детерминированный алгоритм, решающий ту же задачу. Хотя это пока не доказано, имеются серьёзные результаты в пользу этой гипотезы [9]. В работе использованы методы, развитые в [5, 6, 10].

1. Генерические алгоритмы

Пусть I есть множество всех входов некоторой алгоритмической проблемы и I_n — множество всех входов размера n. Для подмножества $S \subseteq I$ определим последовательность

$$\rho_n(S) = \frac{|S \cap I_n|}{|I_n|},\ n = 1, 2, 3, \ldots$$

Заметим, что $\rho_n(S)$ — это вероятность попасть в S при случайной и равновероятной генерации входов из I_n. *Асимптотической плотностью* S назовём предел (если он существует)

$$\rho(S) = \lim_{n\to\infty} \rho_n(S).$$

Множество S называется *генерическим*, если $\rho(S) = 1$, и *пренебрежимым*, если $\rho(S) = 0$. Очевидно, что S генерическое тогда и только тогда, когда его дополнение $I \setminus S$ пренебрежимо.

Алгоритм $\mathcal{A}$ с множеством входов I и множеством выходов $J \cup \{?\}$ $(? \notin J)$ называется *генерическим*, если

1) $\mathcal{A}$ останавливается на всех входах из I;
2) множество $\{x \in I : \mathcal{A}(x) =?\}$ пренебрежимо.

Генерический алгоритм $\mathcal{A}$ вычисляет функцию $f : I \to J$, если для всех $x \in I$ $\mathcal{A}(x) = y \in J \Rightarrow f(x) = y$. Ситуация $\mathcal{A}(x) =?$ означает, что $\mathcal{A}$ не может вычислить функцию f на аргументе x, но условие 2 гарантирует, что $\mathcal{A}$ корректно вычисляет f на почти всех входах (входах из генерического множества).

2. Проблема извлечения корня в группах вычетов

Пусть $\mathbb{Z}/(m)$ — мультипликативная группа вычетов по модулю $m \in \mathbb{N}$. Напомним, что проблема извлечения корня в группах вычетов состоит в вычислении функции $\mathrm{root} : I \to \mathbb{N}$, где I — это множество троек (a, e, m), таких, что $m = pq$ (p, q — различные простые числа), $e < m$, $(\varphi(m), e) = 1$ и $a \in \mathbb{Z}/(m)$. Сама функция root определяется следующим образом:

$$\mathrm{root}(a, e, m) = x \Leftrightarrow x^e = a \in \mathbb{Z}/(m).$$

Под размером входа понимается число разрядов в двоичной записи числа m. Заметим, что при условии $(\varphi(m), e) = 1$ для любого $a \in \mathbb{Z}/(m)$ существует единственный корень степени e [11]. Этим корнем является элемент a^d, где $d = e^{-1} \mod \varphi(m)$. В настоящее время неизвестно полиномиальных алгоритмов (даже вероятностных), решающих

проблему извлечения корня в группах вычетов. На этом факте, в частности, основана криптостойкость известной криптосистемы RSA [7, 8].

Для изучения генерической сложности этой проблемы необходимо провести стратификацию на множестве входов. Рассмотрим любую бесконечную последовательность пар натуральных чисел $\mu = \{(e_1, m_1), (e_2, m_2), \ldots\}$, удовлетворяющую следующим условиям:

1) $2^n < m_n < 2^{n+1}$ для любого n;
2) m_n — произведение двух различных простых чисел для любого $n > 1$;
3) $e_n < m_n$ и $(\varphi(m_n), e_n) = 1$ для любого n.

Будем называть такую последовательность *экспоненциальной*. Из знаменитого постулата Бертрана, доказанного П. Л. Чебышевым, следует, что экспоненциальные последовательности существуют.

Теперь определим функцию root_μ как ограничение функции root на следующее множество входных данных:

$$I = \{(a, e, m) : (e, m) \in \mu, a \in \mathbb{Z}/(m)\}.$$

Под размером входа (a, e, m) понимается количество бит в двоичной записи числа m. Заметим, что множество I_n входов функции root_μ размера n состоит из всех пар (a, e, m), где (e, m) — единственная пара из μ, удовлетворяющая условию $2^n < m < 2^{n+1}$; a — любой элемент из $\mathbb{Z}/(m)$. Таким образом, проблема вычисления функции root_μ является подпроблемой проблемы вычисления функции root. Следующая лемма доказывает, что среди функций root_μ существуют функции, такие же сложные, как и оригинальная функция root.

Лемма 1. Если не существует полиномиального вероятностного алгоритма для вычисления функции root, то найдется такая экспоненциальная последовательность μ, что и для вычисления root_μ нет полиномиального вероятностного алгоритма.

Доказательство. Пусть $P_1, P_2, \ldots$ — все полиномиальные вероятностные алгоритмы. Из предположения о том, что не существует полиномиального вероятностного алгоритма для вычисления root, следует, что для любого алгоритма P_n существует бесконечно много групп $\mathbb{Z}/(m)$, в которых он не может вычислить функцию root для некоторых степеней e. Из этого следует, что можно выбрать последовательность $\mu' = \{(e_1, m_1), (e_2, m_2), \ldots\}$ так, чтобы алгоритм P_n не вычислял root в группе $\mathbb{Z}/(m_n)$ со степенью e_n и для любого n выполнялось бы $m_{n+1} > 2m_n$. Последовательность μ' можно расширить до экспоненциальной последовательности μ, добавив где нужно новые члены. Заметим теперь, что root_μ и есть та функция, для вычисления которой не существует полиномиального алгоритма. ■

3. Основной результат

Следующая теорема говорит о том, что проблема извлечения корня в группах вычетов остается вычислительно трудной и в генерическом случае при условии её труднорешимости в худшем случае.

Теорема 1. Пусть μ — любая экспоненциальная последовательность. Если существует полиномиальный генерический алгоритм, вычисляющий функцию root_μ, то существует полиномиальный вероятностный алгоритм, вычисляющий root_μ для всех входов.

Доказательство. Пусть существует полиномиальный генерический алгоритм $\mathcal{A}$, вычисляющий функцию root_μ. Построим вероятностный полиномиальный алгоритм $\mathcal{B}$,

вычисляющий root_μ на всём множестве входов. Алгоритм $\mathcal{B}$ на входе (a, e, m) работает следующим образом:

1) Сгенерировать случайно и равномерно натуральное $b < m$.
2) Если $(b, m) = 1$, то положить $a' = ab^e$, затем:
   а) запустить алгоритм $\mathcal{A}$ на (a', e, m);
   б) если $\mathcal{A}(a', e, m) = y \in \mathbb{Z}/(m)$, то $a' = y^e = ab^e$, откуда $a = (yb^{-1})^e$ и корень степени e из a равен yb^{-1};
   в) если $\mathcal{A}(a', e, m) =?$, то выдать 1.
3) Если $(b, m) \neq 1$, то $(b, m) = p$, где p — один из двух делителей числа $m = pq$. Таким образом, можно легко посчитать $\varphi(m) = (p-1)(q-1)$ и найти корень как элемент a^d, где $d = e^{-1} \mod \varphi(m)$.

Заметим, что данный полиномиальный вероятностный алгоритм может выдать неправильный ответ только на шаге 2, *в*. Докажем, что вероятность этого меньше $1/2$. Действительно, $a' = ab^e$ при $b \in \mathbb{Z}/(m)$ и $(b, m) = 1$ пробегает все элементы группы $\mathbb{Z}/(m)$, поэтому множество

$$\{(ab^e, e, m) : \ b \in \mathbb{Z}/(m), \ (b, m) = 1\}$$

совпадает с множеством всех входов размера n. Но алгоритм $\mathcal{A}$ генерический, поэтому доля тех входов (a', e, m), на которых он выдаёт неопределённый ответ, стремится к 0 с ростом n и с некоторого момента становится меньше $1/2$. ■

Непосредственным следствием доказанной теоремы является

Теорема 2. Если для вычисления функции root не существует полиномиального вероятностного алгоритма, то существует экспоненциальная последовательность μ, такая, что для вычисления функции root_μ не существует генерического полиномиального алгоритма.

Автор выражает благодарность рецензенту за полезные замечания и предложения по улучшению текста статьи.

ЛИТЕРАТУРА

1. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Generic-case complexity, decision problems in group theory and random walks // J. Algebra. 2003. V. 264. No. 2. P. 665–694.
2. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Average-case complexity for the word and membership problems in group theory // Adv. Math. 2005. V. 190. P. 343–359.
3. *Hamkins J. D. and Miasnikov A. G.* The halting problem is decidable on a set of asymptotic probability one // Notre Dame J. Formal Logic. 2006. V. 47. No. 4. P. 515–524.
4. *Gilman R., Miasnikov A. G., Myasnikov A. D., and Ushakov A.* Report on generic case complexity // Herald of Omsk University. 2007. Special Issue. P. 103–110.
5. *Рыбалов А.* О генерической сложности проблемы распознавания квадратичных вычетов // Прикладная дискретная математика. 2015. № 2(28). С. 54–58.
6. *Рыбалов А.* О генерической сложности проблемы дискретного логарифма // Прикладная дискретная математика. 2016. № 3(33). С. 93–97.
7. *Rivest R., Shamir A., and Adleman L.* A method for obtaining digital signatures and public-key cryptosystems // Commun. ACM. 1978. V. 21. Iss. 2. P. 120–126.
8. *Mao B.* Современная криптография: теория и практика. М.: Вильямс, 2005. 768 с.
9. *Impagliazzo R. and Wigderson A.* P = BPP unless E has subexponential circuits: Derandomizing the XOR Lemma // Proc. 29th STOC. El Paso: ACM, 1997. P. 220–229.

10. *Myasnikov A. and Rybalov A.* Generic complexity of undecidable problems // J. Symbolic Logic. 2008. V. 73. No. 2. P. 656–673.
11. *Коблиц Н.* Курс теории чисел и криптографии. М.: ТВП, 2001. 254 с.

REFERENCES

1. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Generic-case complexity, decision problems in group theory and random walks. J. Algebra, 2003, vol. 264, no. 2, pp. 665–694.
2. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Average-case complexity for the word and membership problems in group theory. Adv. Math., 2005, vol. 190, pp. 343–359.
3. *Hamkins J. D. and Miasnikov A. G.* The halting problem is decidable on a set of asymptotic probability one. Notre Dame J. Formal Logic, 2006, vol. 47, no. 4, pp. 515–524.
4. *Gilman R., Miasnikov A. G., Myasnikov A. D., and Ushakov A.* Report on generic case complexity. Herald of Omsk University, 2007, Special Issue, pp. 103–110.
5. *Rybalov A.* O genericheskoy slozhnosti problemy raspoznavaniya kvadratichnykh vychetov [On generic complexity of the quadratic residuosity problem]. Prikladnaya Diskretnaya Matematika, 2015, no. 2(28), pp. 54–58. (in Russian)
6. *Rybalov A.* O genericheskoy slozhnosti problemy diskretnogo logarifma [On generic complexity of the discrete logarithm problem]. Prikladnaya Diskretnaya Matematika, 2016, no. 3(33), pp. 93–97. (in Russian)
7. *Rivest R., Shamir A., and Adleman L.* A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM, 1978, vol. 21, iss. 2, pp. 120–126.
8. *Mao V.* Sovremennaya kriptografiya: teoriya i praktika [Modern Cryptography: Theory and Practice]. Moscow, Wil'yams Publ., 2005. 768 p. (in Russian)
9. *Impagliazzo R. and Wigderson A.* P = BPP unless E has subexponential circuits: Derandomizing the XOR Lemma. Proc. 29th STOC, El Paso, ACM, 1997, pp. 220–229.
10. *Myasnikov A. and Rybalov A.* Generic complexity of undecidable problems. J. Symbolic Logic, 2008, vol. 73, no. 2, pp. 656–673.
11. *Koblits N.* Kurs teorii chisel i kriptografii [Course of Number Theory and Cryptography]. Moscow, TVP Publ., 2001. 254 p. (in Russian)