МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

УДК 510.52 DOI 10.17223/20710410/67/6

О ГЕНЕРИЧЕСКОЙ СЛОЖНОСТИ РЕШЕНИЯ УРАВНЕНИЙ НАД БИЦИКЛИЧЕСКИМ МОНОИДОМ[1]

А. А. Лопатин*, А. Н. Рыбалов**

*Государственный университет города Кампинас, г. Кампинас, Сан-Паулу, Бразилия
**Институт математики им. С. Л. Соболева СО РАН, г. Омск, Россия

E-mail: dr.artem.lopatin@gmail.com, alexander.rybalov@gmail.com

Изучается вычислительная сложность проблемы проверки разрешимости систем уравнений над бициклическим моноидом. Этот моноид, помимо теоретического значения в топологии и теории полугрупп, имеет приложения в информатике и языках программирования, например как модель для языка Дика сбалансированных скобочных выражений. Доказывается NP-полнота проблемы проверки разрешимости систем уравнений над бициклическим моноидом. Также доказывается, что при $P \neq NP$ и $P = BPP$ для этой проблемы не существует сильно генерического полиномиального алгоритма. Это означает, что для любого генерического полиномиального алгоритма имеется эффективный метод случайной генерации входов, на которых этот алгоритм не может решить рассматриваемую проблему. Полученный результат указывает на возможное применение данной проблемы в криптографии, где нужно, чтобы проблема взлома криптосистемы была трудной для почти всех входов.

Ключевые слова: *генерическая сложность, NP-полнота, бициклический моноид.*

ON GENERIC COMPLEXITY OF EQUATION SOLVING OVER THE BICYCLIC MONOID

A. A. Lopatin*, A. N. Rybalov**

*Universidade Estadual de Campinas (UNICAMP), Brazil
**Sobolev Institute of Mathematics, Omsk, Russia

In this paper, we study computational complexity of the problem of determining solvability equations over bicyclic monoid. This monoid, in addition to its theoretical significance in topology and semigroup theory, has applications in computer science and programming languages, for example, as a model for the Dyck language of balanced bracket expressions. We prove NP-completeness of the problem of determining solvability equations over bicyclic monoid. Also, we prove that if $P \neq NP$ and $P = BPP$, then for this problem there is no strongly generic polynomial algorithm. This means

[1]Первый автор поддержан грантом FAPESP 2023/17918-2. Работа второго автора выполнена в рамках госзадания ИМ СО РАН, проект FWNF-2022-0003.

that for any generic polynomial algorithm there exists an efficient method of randomly generating inputs on which the algorithm cannot solve the problem under consideration. The result points to a possible application of this problem in cryptography, where it is necessary that the problem of breaking a cryptosystem be hard for almost all inputs. To prove this theorem, we use the method of generic amplification, which allows to construct generically hard problems from the problems hard in the classical sense. The basis of this method is a technique of cloning, which combines the input data of a problem into sufficiently large sets of equivalent input data. Equivalence is understood in the sense that the problem is solved similarly for them.

Keywords: *generic complexity, NP-completeness, bicyclic monoid.*

Введение

Решение уравнений и систем уравнений над вещественными, комплексными, рациональными, целыми числами является классической темой исследований в различных областях математики в течение тысяч лет. Классическая алгебраическая геометрия изучает множества решений алгебраических уравнений над полями вещественных и комплексных чисел. В рамках диофантовой геометрии и диофантова анализа изучаются решения алгебраических уравнений над целыми и рациональными числами. В XX в. большую роль начали играть вычислительные аспекты этих теорий. Изучение алгоритмических проблем, связанных с определением наличия решения у систем уравнений, а также с нахождением и описанием множества решений, является темой многочисленных теоретических и практических исследований.

В последние десятилетия фокус исследований перемещается на неклассические области, такие, как группы [1], полугруппы [2, 3], графы [4], частичные порядки [5]. Потребность решения уравнений в этих системах возникает при рассмотрении различных практических проблем информатики, криптографии, теории языков программирования. Например, свободные полугруппы являются базисом для описания важнейших классов формальных языков и грамматик: регулярных, контекстно свободных. Часто при этом изучаемый формальный язык задаётся некоторым набором уравнений, множество решений которых даёт нужный язык. К необходимости решения уравнений над графами приводят задачи проверки вложимости (совместимости) одной коммуникационной сети в другую сеть.

К сожалению, как правило, проблема решения систем уравнений над различными алгебраическими системами является либо неразрешимой, либо имеет большую вычислительную сложность. Даже над конечными алгебраическими системами эта проблема оказывается NP-полной. Это означает, что при условии $\mathrm{P} \neq \mathrm{NP}$ для неё не существует полиномиальных алгоритмов. Поэтому актуальным является изучение генерической сложности [6] данных проблем. В рамках генерического подхода алгоритмическая проблема рассматривается не на всём множестве входов, а на некотором подмножестве «почти всех» входов. С одной стороны, положительные результаты о возможности эффективного решения каких-либо трудных задач для почти всех входов полезны для практики. С другой стороны, негативные результаты о генерической трудности некоторых проблем дают надежду на возможное их использование в криптографии, где как раз важно, чтобы проблема взлома криптосистемы была трудной для почти всех входов. Генерическая сложность проблем решения уравнений над конечными полями и полугруппами изучалась в [7].

В данной работе рассматривается проблема разрешимости уравнений над бициклическим моноидом $B = \langle a, b \,|\, ab = e \rangle$. Этот моноид, помимо теоретического значения

в топологии и теории полугрупп, имеет приложения в информатике и языках программирования, например, как модель для так называемого языка Дика сбалансированных скобочных выражений [8]. Что касается проблемы разрешимости уравнений над бициклическим моноидом, то известно, что она разрешима [9]. Однако вопрос о вычислительной сложности этой проблемы пока не изучался. В настоящей работе исследуется вычислительная сложность проблемы проверки разрешимости систем уравнений над бициклическим моноидом.

1. Предварительные сведения

Бициклическим моноидом будем называть моноид, заданный конечно определённым представлением $B = \langle a, b \,|\, ab = e\rangle$, где e — пустое слово, выполняющее роль единицы, с операцией умножения — конкатенацией слов. *Нормальной формой* слова w над алфавитом $\{a, b\}$ в B назовём слово $b^m a^n$. Очевидно, любой элемент моноида B можно привести к такой нормальной форме. Под *размером* элемента $b^m a^n$ будем понимать сумму длин двоичных записей чисел m и n. Легко подсчитать, что нормальной формой произведения двух произвольных элементов $x = b^m a^n$ и $y = b^k a^l$ в бициклическом моноиде B является

$$x \cdot y = b^m a^n \cdot b^k a^l = b^{m+k-\min\{n,k\}} \cdot a^{l+n-\min\{n,k\}}.$$

Уравнение от переменных $x_1, \ldots, x_n$ над бициклическим моноидом B имеет вид

$$c_1 x_1 c_2 \ldots c_m x_m c_{m+1} = c_{m+2} x_{m+1} c_{m+3} \ldots c_{n+1} x_n c_{n+2},$$

где $c_1, \ldots, c_{n+2} \in B$. Набор элементов $d_1, \ldots, d_n$ является *решением* этого уравнения, если при его подстановке вместо соответствующих переменных получается верное равенство. Под *размером* уравнения будем понимать сумму размеров элементов $c_1, \ldots, c_{n+2}$ плюс число переменных n. *Системой уравнений* называется конечный набор уравнений. *Решением* системы называется такой набор элементов, который является одновременно решением всех уравнений данной системы. Под *размером* системы будем понимать сумму размеров всех её уравнений. *Проблема совместности систем уравнений над бициклическим моноидом* состоит в следующем: по произвольной заданной системе уравнений над B определить, имеет ли она решение в B.

Напомним, что алгоритмическая проблема распознавания $A \subseteq I$ принадлежит *классу* NP, если существует полиномиальный алгоритм $\mathcal{A}$ и полином $p(n)$, такие, что

$$x \in A \Leftrightarrow \exists y \in I \ \big(|y| < p(|x|) \ \&\ \mathcal{A}(x, y) = 1\big).$$

Здесь через $|x|$ обозначена длина входа x. Элемент y называют подсказкой и говорят, что алгоритм $\mathcal{A}$ проверяет эту подсказку.

Алгоритмическая проблема распознавания $A \subseteq I$ называется NP-*трудной*, если к ней за полиномиальное время сводится любая проблема $B \subseteq I$ из класса NP. То есть существует функция $f : I \to I$, вычислимая некоторым полиномиальным алгоритмом, такая, что

$$x \in B \Leftrightarrow f(x) \in A.$$

Если при этом NP-трудная проблема сама принадлежит классу NP, то она называется NP-*полной*.

Основные определения генерического подхода можно найти в [6]. Определения вычислительных классов P, NP и BPP содержатся в [10].

2. NP-полнота

Нетрудно показать, что любую систему уравнений над бициклическим моноидом можно привести к эквивалентной системе, в которой каждое уравнение является равенством одного из следующих типов:

1) $x_i = x_j x_k$;
2) $x_i = a$;
3) $x_i = b$.

Будем называть такие системы уравнений *системами в форме Сколема*. Размер такой системы — это число уравнений в ней. Заметим, что при таком переходе размер системы в форме Сколема по сравнению с размером исходной системы увеличивается не более чем полиномиально.

Теорема 1. Проблема проверки совместности систем уравнений над бициклическим моноидом является NP-полной.

Доказательство. Докажем сначала принадлежность проблемы классу NP. По заданной системе уравнений S над B построим эквивалентную ей систему в форме Сколема S'. Затем по системе S' построим бесквантерную формулу $\Phi_{S'}$ арифметики Пресбургера (то есть над натуральными числами со сложением, вычитанием и предикатом порядка $<$), такую, что выполнимость $\Phi_{S'}$ равносильна существованию решения системы S. Для этого каждой переменной x системы S' сопоставим пару натуральнозначных переменных (y, z), которая соответствует нормальной форме $x = b^y a^z$. Каждому уравнению системы S' сопоставим формулу арифметики Пресбургера следующим образом:

1) если уравнение имеет вид $x_i = x_j x_k$, то соответствующая формула будет такой:

$$\Big((y_k \leqslant z_j)\&(y_i = y_j)\&(z_i = z_j - y_k + z_k)\Big) \vee \Big((z_j < y_k)\&(y_i = y_k - z_j + y_j)\&(z_i = z_k)\Big).$$

   Эта формула соответствует правилу изменения степеней букв a и b в нормальной форме элементов бициклического моноида при перемножении $x_j = b^{y_j} a^{z_j}$ на $x_k = b^{y_k} a^{z_k}$;
2) если уравнение имеет вид $x_i = a$, то соответствующая формула будет такой:

$$(y_i = 0)\&(z_i = 1);$$

3) если уравнение имеет вид $x_i = b$, то соответствующая формула будет такой:

$$(y_i = 1)\&(z_i = 0).$$

Итоговая формула $\Phi_{S'}$ есть коньюнкция формул, соответствующих всем уравнениям системы S'. Из построения формулы $\Phi_{S'}$ следует, что система S' совместна над B тогда и только тогда, когда формула $\Phi_{S'}$ выполнима в арифметике Пресбургера. Из свойств целочисленных решений систем линейных неравенств (см., например, [11]) следует, что если $\Phi_{S'}$ выполнима, то размер в двоичной записи минимальных значений переменных, делающих истинной формулу $\Phi_{S'}$, ограничен полиномиально от длины $\Phi_{S'}$. Таким образом, в качестве подсказки (решения) можно взять значения этих переменных. Это доказывает принадлежность проблемы совместности систем уравнений в бициклическим моноиде классу NP.

Докажем теперь, что к проблеме совместности систем уравнений в бициклическим моноиде полиномиально сводится известная NP-полная проблема о выполнимости

3-КНФ, которая заключается в следующем: 3-КНФ $\Phi(x_1, \ldots, x_n)$ — это конъюнкция дизъюнкций вида $(\alpha_1 \vee \alpha_2 \vee \alpha_3)$, где α_i, $i = 1, 2, 3$, есть либо булева переменная x_k, либо отрицание булевой переменной $\overline{x_k}$. Нужно определить, является ли заданная 3-КНФ $\Phi(x_1, \ldots, x_n)$ выполнимой, то есть существуют ли значения $a_1, \ldots, a_n \in \{0, 1\}^n$, такие, что $\Phi(a_1, \ldots, a_n) = 1$.

Построим по 3-КНФ $\Phi(x_1, \ldots, x_n)$ систему уравнений над бициклическим моноидом, которая имеет решение тогда и только тогда, когда $\Phi(x_1, \ldots, x_n)$ выполнима. Для этого каждой булевой переменной из Φ сопоставим две переменные x и y, а также запишем уравнения

$$xa = ax,$$
$$ya = ay,$$
$$xy = a.$$

Первые два уравнения гарантируют, что возможные значения для x и y принадлежат подмоноиду, порождённому a, а третье — что x, y могут принимать только значения a или e. Элемент a выполняет роль логической 1, а e — роль логического 0. У самих переменных роли таковы: x моделирует соответствующую логическую переменную, а y — её отрицание.

Далее смоделируем дизъюнкцию $(\alpha_1 \vee \alpha_2 \vee \alpha_3)$. Для этого возьмём переменные z_1, z_2, z_3 над бициклическим моноидом, соответствующие переменным (или их отрицаниям) из дизъюнкции, и запишем уравнения

$$z_1 z_2 z_3 = au,$$
$$ua = au,$$

где u — некоторая новая переменная. Заметим, что эта система уравнений разрешима в бициклическом моноиде тогда и только тогда, когда хотя бы одна из переменных z_1, z_2, z_3 равна a, что соответствует истинности соответствующей дизъюнкции.

Теперь по каждой дизъюнкции 3-КНФ $\Phi(x_1, \ldots, x_n)$ строим подобные уравнения и получаем систему S_Φ, которая имеет решения в бициклическом моноиде тогда и только тогда, когда 3-КНФ $\Phi(x_1, \ldots, x_n)$ выполнима. Полиномиальность данной сводимости следует из описанного процесса построения. ■

3. Генерическая трудноразрешимость

Будем называть систему в форме Сколема *нормализованной*, если в k-м уравнении системы могут встречаться только переменные x_i, где $i \leqslant 3k$. Очевидно, что любую систему в форме Сколема можно нормализовать с помощью подходящей перенумерации переменных. Обозначим через $\mathcal{S}$ множество всех нормализованных систем уравнений в форме Сколема.

Лемма 1. Число нормализованных систем уравнений в форме Сколема размера n равно

$$|\mathcal{S}_n| = \prod_{k=1}^{n} (27k^3 + 6k). \tag{1}$$

Доказательство. Для t-го уравнения в системе $S \in \mathcal{S}_n$ существует $(3t)^3$ вариантов выбрать уравнение вида $x_i = x_j x_k$ и $6t$ вариантов выбрать уравнение вида $x_i = a$ или $x_i = b$. Итого для t-го уравнения есть $27t^3 + 6t$ вариантов и для системы из n уравнений выполняется (1). ■

Для произвольной системы уравнений $S = \{S_1, \ldots, S_m\}$ рассмотрим множество систем $eq(S)_n$, которые получаются добавлением к системе S произвольных уравнений $S_{m+1}, \ldots, S_n$, где l-е уравнение имеет вид $x_i = x_j x_k$, причём $3m < i, j, k < 3(l+m)$. Очевидно, что любая система из $eq(S)_n$ совместна над B тогда и только тогда, когда совместна над B система S.

Лемма 2. Для любой системы S размера m и любого $n > m$ имеет место оценка

$$\frac{|eq(S)_n|}{|\mathcal{S}_n|} > \frac{1}{2(33n^3)^m}.$$

Доказательство. Пусть $n > m$. Для t-го добавленного к S уравнения вида $x_i = x_j x_k$, где $3m < i, j, k < 3(t+m)$, имеется $(3t)^3 = 27t^3$ вариантов. Поэтому

$$|eq(S)_n| = \prod_{t=1}^{n-m} (27t^3).$$

Теперь по лемме 1 получим

$$\rho_n(eq(S)_n) = \frac{|eq(S)_n|}{|\mathcal{S}_n|} = \frac{\prod\limits_{k=1}^{n-m} (27k^3)}{\prod\limits_{k=1}^{n} (27k^3 + 6k)} = \prod_{k=1}^{n-m} \Big(\frac{27k^2}{27k^2+6}\Big) \prod_{k=n-m+1}^{n} \frac{1}{27k^3+6k}.$$

Оценим снизу первое произведение:

$$\prod_{k=1}^{n-m} \Big(\frac{27k^2}{27k^2+6}\Big) = \prod_{k=1}^{n-m} \Big(1 - \frac{1}{9k^2+2}\Big) > \prod_{k=2}^{n-m+1} \Big(1 - \frac{1}{k^2}\Big) = \prod_{k=2}^{n-m+1} \frac{(k-1)(k+1)}{k^2} =$$
$$= \frac{1\cdot 3}{2^2} \frac{2\cdot 4}{3^2} \cdot \ldots \cdot \frac{(n-m-1)(n-m+1)}{(n-m)^2} \frac{(n-m)(n-m+2)}{(n-m+1)^2} = \frac{n-m+2}{2(n-m+1)} > \frac{1}{2}.$$

Теперь оценим второе произведение:

$$\prod_{k=n-m+1}^{n} \frac{1}{27k^3+6k} > \frac{1}{(27n^3+6n)^m}.$$

Итого получаем $\rho_n(eq(S)) = \dfrac{|eq(S)_n|}{|\mathcal{S}_n|} > \dfrac{1}{2(27n^3+6n)^m} > \dfrac{1}{2(33n^3)^m}$. ■

Теорема 2. Пусть P $\neq$ NP и P = BPP. Тогда для проблемы проверки совместности нормализованных систем уравнений в форме Сколема над бициклическим моноидом не существует сильно генерического полиномиального алгоритма.

Доказательство. Допустим, что существует сильно генерический полиномиальный алгоритм $\mathcal{A}$, определяющий совместность систем уравнений над B. Построим вероятностный полиномиальный алгоритм $\mathcal{B}$, решающий эту проблему на всём множестве входов. На системе S размера n алгоритм $\mathcal{B}$ работает следующим образом:

1) генерирует случайно и равновероятно систему S' из множества $eq(S)$ размера n^2;
2) запускает алгоритм $\mathcal{A}$ на системе S';
3) если $\mathcal{A}(S') \neq ?$, то алгоритм правильно определяет, совместна ли система S', а вместе с ней и система S;
4) если $\mathcal{A}(S') = ?$, то выдаёт ответ «НЕТ».

Заметим, что полиномиальный вероятностный алгоритм $\mathcal{B}$ выдаёт правильный ответ на шаге 3, а на шаге 4 может выдать неправильный ответ. Надо доказать, что вероятность того, что ответ выдаётся на шаге 4, меньше $1/3$.

Оценим вероятность выдачи ответа на шаге 4. Вероятность того, что для S' имеет место $\mathcal{A}(S') = ?$, не больше

$$\frac{|\{S' \in \mathcal{S} : \mathcal{A}(S') = ?\}_{n^2}|}{|eq(S)_{n^2}|} = \frac{|\{S' \in \mathcal{S} : \mathcal{A}(S') = ?\}_{n^2}|}{|\mathcal{S}_{n^2}|} \frac{|\mathcal{S}_{n^2}|}{|eq(S)_{n^2}|}.$$

Так как множество $\{S' \in \mathcal{S} : \mathcal{A}(S') = ?\}$ сильно пренебрежимое, то существует константа $\alpha > 0$, такая, что

$$\frac{|\{S' \in \mathcal{S} : \mathcal{A}(S') = ?\}_{n^2}|}{|\mathcal{S}_{n^2}|} < \frac{1}{2^{\alpha n^2}}$$

для любого n. По лемме 2

$$\frac{|\mathcal{S}_{n^2}|}{|eq(S)_{n^2}|} < 2(33n^6)^n.$$

Поэтому искомая вероятность ответа на шаге 4 не больше

$$\frac{2(33n^6)^n}{2^{\alpha n^2}} = \frac{2^{1+\log 33n+6n\log n}}{2^{\alpha n^2}} = \frac{1}{2^{\alpha n^2-6n\log n-\log 33n-1}} < \frac{1}{3}$$

при достаточно больших n.

Таким образом, проблема проверки совместности систем уравнений над B принадлежит классу BPP. Так как BPP = P, она принадлежит классу P. Это, по теореме 1, противоречит условию P $\neq$ NP. ■

Авторы выражают искреннюю благодарность рецензенту за полезные замечания и предложения по улучшению текста статьи.

ЛИТЕРАТУРА

1. *Baumslag G., Myasnikov A., and Remeslennikov V.* Algebraic geometry over groups I. Algebraic sets and ideal theory // J. Algebra. 1999. V. 219. No. 1. P. 16–79.
2. *Shevlyakov A. N.* Equationally Noetherian varieties of semigroups and B. Plotkin's problem // Сиб. электрон. матем. изв. 2023. Т. 20. № 2. С. 724–734.
3. *Шевляков А. Н.* Сплетения полугрупп и проблема Б. И. Плоткина // Алгебра и логика. 2023. Т. 62. № 5. С. 665–691.
4. *Рыбалов А. Н.* О сложности решения уравнений над графами // Сиб. электрон. матем. изв. 2024. Т. 21. № 1. С. 62–69.
5. *Nikitin A. Yu. and Shevlyakov A. N.* On radicals over strict partial order sets // J. Physics: Conf. Ser. 2021. V. 1791. No. 012080. P. 1–6.
6. *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.
7. *Rybalov A. and Shevlyakov A.* Generic complexity of solving of equations in finite groups, semigroups and fields // J. Physics: Conf. Ser. 2021. V. 1901. No. 012047. P. 1–8.
8. *Hopcroft J. E. and Ullman J. D.* Formal Languages and Their Relation to Automata. Addison-Wesley, 1969. 288 p.
9. *Diekert V. and Lohrey M.* Word equations over graph products // Intern. J. Algebra Computation. 2008. V. 18. No. 3. P. 493–533.

10. *Китаев А., Шень А., Вялый М.* Классические и квантовые вычисления. М.: МЦНМО, ЧеРо, 1999. 192 с.
11. *Схрейвер А.* Теория линейного и целочисленного программирования. М.: Мир, 1991. 360 с.

REFERENCES

1. *Baumslag G., Myasnikov A., and Remeslennikov V.* Algebraic geometry over groups I. Algebraic sets and ideal theory. J. Algebra, 1999, vol. 219, no. 1, pp. 16–79.
2. *Shevlyakov A. N.* Equationally Noetherian varieties of semigroups and B. Plotkin's problem. Siberian Electronic Math. Reports, 2023, vol. 20, no. 2, pp. 724–734.
3. *Shevlyakov A. N.* Spleteniya polugrupp i problema B. I. Plotkina [Wreath products of semigroups and Plotkin's problem]. Algebra i Logika, 2023, vol. 62, no. 5, pp. 665–691. (in Russian)
4. *Rybalov A. N.* O slozhnosti resheniya uravneniy nad grafami [On complexity of solving of equations over graphs]. Siberian Electronic Math. Reports, 2024, vol. 21, no. 1, pp. 62–69. (in Russian)
5. *Nikitin A. Yu. and Shevlyakov A. N.* On radicals over strict partial order sets. J. Physics: Conf. Ser., 2021, vol. 1791, no. 012080, pp. 1–6.
6. *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.
7. *Rybalov A. and Shevlyakov A.* Generic complexity of solving of equations in finite groups, semigroups and fields. J. Physics: Conf. Ser., 2021, vol. 1901, no. 012047, pp. 1–8.
8. *Hopcroft J.E. and Ullman J.D.* Formal Languages and Their Relation to Automata. Addison-Wesley, 1969. 288 p.
9. *Diekert V. and Lohrey M.* Word equations over graph products. Intern. J. Algebra Computation, 2008, vol. 18, no. 3, pp. 493–533.
10. *Kitaev A., Shen' A., and Vyalyy M.* Klassicheskie i kvantovye vychisleniya [Classical and Quantum Computations]. Moscow, MCCME Publ., 1999. 192 p. (in Russian)
11. *Schrijver A.* Theory of Linear and Integer Programming. Wiley, 1998. 484 p.