Сложность вычислений

Вероятностная проверка на простоту без ошибок

Чернис Константин, группа 694

Содержание

В данном проекте доказываются избранные факты вероятностной проверки чисел на простоту, а также проводятся некоторые эксперименты.

1. Введение в сложностные классы

Для начала опишем сложностные классы, затрагиваемые данной задачей:

Определение 1.1. Вероятностной машиной Тьюринга называется детерминированная машина Тьюринга M с двумя аргументами x (аргумент вероятностной машины) и r (случайные биты), где длина r есть некоторая функция от длины x. Результатом работы M на входе x будет вероятностое распределение, индуцированное данным x и равномерным на всех значениях r. Временем работы M на данном x будем считать максимальное время работы $M(x, r)$ для всех r указанной длины. Так же определяется и использованная память.

Определение 1.2. Классом **RP** называется класс языков A, для которых существует полиномиальный в худшем случае вероятностный алгоритм V, такой что:

- если $x \in A$, то $P_r[V(x, r) = 1] \geqslant \frac{1}{2}$;
- если $x \notin A$, то $P_r[V(x, r) = 1] = 0$.

Определение 1.3. Классом **coRP** называется класс языков A, для которых существует полиномиальный в худшем случае вероятностный алгоритм V, такой что:

- если $x \in A$, то $P_r[V(x, r) = 1] = 1$;
- если $x \notin A$, то $P_r[V(x, r) = 1] \leqslant \frac{1}{2}$.

Определение 1.4. Классом **ZPP** называется класс языков A, для которых существует вероятностный алгоритм A, такой что

$$x \in A \iff \forall r\ V(x, r) = 1,$$

а для каждого x ожидаемое по r время работы полиномиально.

Обозначение **ZPP** расшифровывается как "zero-error probabistic polynomial".

Утверждение 1.1. $\mathbf{ZPP} = \mathbf{RP} \cap \mathbf{coPR}$.

Таким образом, для вероятностной проверки чисел на простоту достаточно предоставить алгоритмы проверки чисел на простоту из **RP** и **coRP**, после чего запускать их по очереди до тех пор, пока один из алгоритмов не выдаст ответ, в котором он уверен. Вероятность отсутствия ответа будет уменьшаться минимум в 4 раза после каждой итерации цикла проверки, так что за полиномиальное число шагов вероятность станет экспоненциально малой и можно будет применить детерминированный экспоненциальный алгоритм.

2. Сертификаты

Утверждение 2.1. RP $\subset$ NP

Доказательство. Действительно, любое значение r, при котором $V(x, r) = 1$, будет доказательством того, что $x \in A$. ∎

Следствие 2.1. coRP $\subset$ coNP

Таким образом, в силу того, что язык простых чисел, как будет показано далее, лежит в **ZPP**, для любого $n \in \mathbb{Z}$ существует либо сертификат простоты, либо сертификат того, что n составное, проверяемый за полиномиальное время. Таким образом, единожды проверив число на простоту за вероятностно полиномиальное время, в дальнейшем можно снова доказать корректность проверки уже за детерминированный полином. Эти сертификаты будут указаны для каждого описанного ниже алгоритма соответственно.

В следующей секции будет описан алгоритм из **coRP**, а в секции 4 — из **RP**.

3. Алгоритм Миллера-Рабина

Большинство алгоритмов вероятностной проверки на простоту из **coRP** опираются на какое-либо свойство простых чисел, то есть проверяют необходимое условие. Наиболее популярным среди них является алгоритм Миллера-Рабина, который гарантирует, что для нечётного составного минимум 75% чисел от 1 до $n-1$ позволяют определить его непростоту.

Говоря в терминах Определения 1.3, A — множество простых чисел, и для $x \notin A$ $P_r[V(x, r) = 1] \leqslant \dfrac{1}{4}$, где $r \in \overline{1, n-1}$. Кроме того, как будет показано ниже, проверяемое условие действительно является необходимым, то есть для $x \in A$ $P_r[V(x, r) = 1] = 1$, то есть алгоритм Миллера-Рабина лежит в **coNP**.

3.1 Описание алгоритма

Заданное нечётное целое число $n > 1$ можно представить в виде $n - 1 = 2^e k$, где $e \geqslant 1$ (т.к. n нечётно) и k нечётное. Применяя к $x^{n-1} - 1 = x^{2^e k} - 1$ формулу разности квадратов, получаем:

$$\begin{aligned} x^{2^e k} - 1 &= \left(x^{2^{e-1} k}\right)^2 - 1 \\ &= \left(x^{2^{e-1} k} - 1\right)\left(x^{2^{e-1} k} + 1\right) \\ &= \left(x^{2^{e-2} k} - 1\right)\left(x^{2^{e-2} k} + 1\right)\left(x^{2^{e-1} k} + 1\right) \\ &\vdots \\ &= \left(x^k - 1\right)\left(x^k + 1\right)\left(x^{2k} + 1\right)\left(x^{4k} + 1\right) \ldots \left(x^{2^{e-1} k} + 1\right) \end{aligned}$$

Если n простое и $a \in \overline{1, n-1}$, то по малой теореме Ферма $a^{n-1} - 1 \equiv 0 \bmod n$. Используя разложение, полученное выше, имеем

$$\left(x^k - 1\right)\left(x^k + 1\right)\left(x^{2k} + 1\right)\left(x^{4k} + 1\right) \ldots \left(x^{2^{e-1}k} + 1\right) \equiv 0 \bmod n$$

Таким образом, для простого n один из множителей должен делиться на n, то есть необходимым условием, нарушение которого означает, что число составное, является

$$a^k \equiv 1 \bmod n \text{ или } a^{2^i k} \equiv -1 \bmod n \text{ для некоторого } i \in \overline{0, e-1}.$$

Определение 3.1. Представим нечётное $n > 1$ в виде $n - 1 = 2^e k$, где e нечётно и выберем $a \in \overline{1, n-1}$. Тогда a называется свидетелем для числа n, если не выполнено необходимое условие, то есть

$$a^k \not\equiv 1 \bmod n \text{ и } a^{2^i k} \not\equiv -1 \bmod n \; \forall i \in \overline{0, e-1}.$$

Если же необходимое условие выполнено, то есть

$$a^k \equiv 1 \bmod n \text{ или } a^{2^i k} \equiv -1 \bmod n \text{ для некоторого } i \in \overline{0, e-1},$$

то a не является свидетелем для n.

Отметим, что уже сейчас можно построить вероятностный алгоритм проверки на простоту со сколь угодно малой вероятностью ошибки:

Data: проверяемое число n, количество итераций t
Result: является ли число n простым
for $i \in \overline{1, t}$ **do**
  выбрать случайное a из $\overline{1, n-1}$;
  if *a является свидетелем для n* **then**
    return "n составное";
  end
end
return "n простое с вероятностью минимум $1 - 1/4^t$";

3.2 Доказательство оценки на число свидетелей

Для начала покажем, что оценка 75% неулучшаема:

Утверждение 3.1. Доля свидетелей для $n = 9$ составляет $3/4$.

Доказательство. $n - 1 = 8 = 2^3$, так что $e = 3$ и $k = 1$, и для проверки необходимого условия надо перебрать (a, a^2, a^3). Из приведённой ниже таблицы видно, что свидетелями среди $\overline{1, 8}$ являются $2, 3, 4, 5, 6, 7$, что составляет $6/8 = 3/4$, что и требовалось.

$a \bmod 9$	1	2	3	4	5	6	7	8
$a^2 \bmod 9$	1	4	0	7	7	0	4	1
$a^3 \bmod 9$	1	7	0	4	4	0	7	1

■

Существует также доказательство неулучшаемости оценки при $n \to \infty$, оно приведено в [3].

Теорема 3.1. Пусть $n > 1$ нечётное составное.

Доля целых чисел среди $\overline{1, n-1}$, являющихся свидетелями числа n, превышает 75%, за исключением $n = 9$, для которого доля составляет 75%.

Другими словами, доля целых чисел среди $\overline{1, n-1}$, не являющихся свидетелями числа n, меньше 25%, за исключением $n = 9$, для которого доля составляет 25%.

Докажем более слабое утверждение:

Теорема 3.2. Если $n > 1$ нечётное и составное, то доля свидетелей числа n превышает 50%. Другими словами, больше 50% из $a \in \overline{1, n-1}$ удовлетворяют $a^k \not\equiv 1 \bmod n$ и $a^{2^i k} \not\equiv -1 \bmod n \; \forall i \in \overline{0, e-1}$.

Доказательство. Докажем, что доля не свидетелей для n меньше 50%, показав, что они образуют собственную подгруппу группы обратимых чисел mod n. В силу того, что порядок собственной подгруппы составляет максимум половину от порядка группы, множество свидетелей числа n содержит минимум половину обратимых чисел mod n и все необратимые числа mod n среди $\overline{1, n-1}$ (множество необратимых непусто в силу того, что n составное). Таким образом, доля свидетелей для числа n первышает 50%.

Случай 1: n является степенью простого числа, то есть $n = p^\alpha$, где p — нечётное простое и $\alpha \geqslant 2$.

Утверждение 3.2. Если $n = p^\alpha$ для простого p и $\alpha \geqslant 1$, то не свидетели для n являются корнями уравнения $a^{p-1} \equiv 1 \bmod p^\alpha$, которые образуют группу по умножению mod n.

Доказательство. Обоснование приведено в [2]. ■

Согласно Утверждению 2.2 свидетели непростоты образуют группу по умножению mod n. Порядок числа a, являющегося решением уравнения $a^{p-1} \equiv 1 \bmod n$, делит $p - 1$, так что он не делится на p. В то же время существуют обратимые mod n числа, порядок которых делится на p: примером такого числа является $1 + p$, чей порядок mod p^α составляет $p^{\alpha-1}$ (этот факт можно показать индукцией по r: база — $1 + kp \equiv 1 \bmod p$, переход — $(1 + kp^r)^p \equiv 1 \bmod p^{r+1}$). Таким образом, не свидетели mod n образуют собственную подгруппу в группе обратимых чисел mod n, что заканчивает доказательство этого случая.

Случай 2: n не является степенью простого. Пусть $i_0 \in \overline{0, e-1}$ — максимальное число, такое что $\exists\, a_0 \in \mathbb{Z}$ такой что $a^{2^{i_0}} \equiv -1 \bmod n$. (В силу того, что $(-1)^{2^0} = -1$, требуемый i_0 существует, причём a_0 взаимно прост с n).

Множество

$$G_n = \{\, a \in \overline{1, n-1} \mid a^{2^{i_0} k} \equiv \pm 1 \bmod n \,\}$$

является группой по умножению mod n и содержит все a, удовлетворяющие одному из двух условий:

(1) $a^k \equiv 1 \bmod n$,

(2) $a^{2^i k} \equiv 1 \bmod n$ для одного из $i \in \overline{0, e-1}$.

Если $a^k \equiv 1 \bmod n$, то $a^{2^{i_0} k} \equiv 1 \bmod n$. Если же $a^{2^i k} \equiv 1 \bmod n$ для некоторого $i \in \overline{0, e-1}$, то $\left(2^k\right)^{2^i} \equiv -1 \bmod n$, причём $i \leqslant i_0$ в силу максимальности i_0. Таким образом, $a^{2^{i_0}} \equiv -1 \bmod n$, если $i = i_0$, и $a^{2^{i_0}} = 1 \bmod n$, если $i < i_0$. Отсюда все $a \in \overline{1, n-1}$, удовлетворяющие (1) или (2), лежат в G_n.

Покажем, что G_n является собственной подгруппой обратимых чисел mod n, для чего найдём обратимое число, не лежащее в G_n. Пусть p — простой делитель n, тогда представим n в виде $n = p^\alpha n'$, где $\alpha \geqslant 1$ и $p \nmid n'$. p^α и n' нечётные и не равны 1 (в силу того, что n не является степенью простого) $\implies p^\alpha, n' \geqslant 3$.

Согласно китайской теореме об остатках, $\exists\, a \in \overline{1, n-1}$, удовлетворяющий следующим двум уравнениям:

$$a \equiv a_0 \bmod p^\alpha, \quad a \equiv 1 \bmod n'.$$

Выше показали, что $(a_0, n) = 1 \implies (a, n) = 1$ (т.к. $(a, n') = 1$), то есть a является обратимым mod n. Тогда для доказательства того, что подгруппа G_n не является собственной, остаётся показать, что $a \notin G_n$.

$$a^{2^{i_0} k} \equiv a_0^{2^{i_0} k} \equiv (-1)^k \equiv -1 \bmod p^\alpha \implies a^{2^{i_0} k} \not\equiv 1 \bmod n$$

в силу того, что $-1 \not\equiv 1 \bmod p^\alpha$ (т.к. $p^\alpha \geqslant 3$). Кроме того,

$$a^{2^{i_0} k} \equiv 1 \bmod n' \implies a^{2^{i_0} k} \not\equiv -1 \bmod n$$

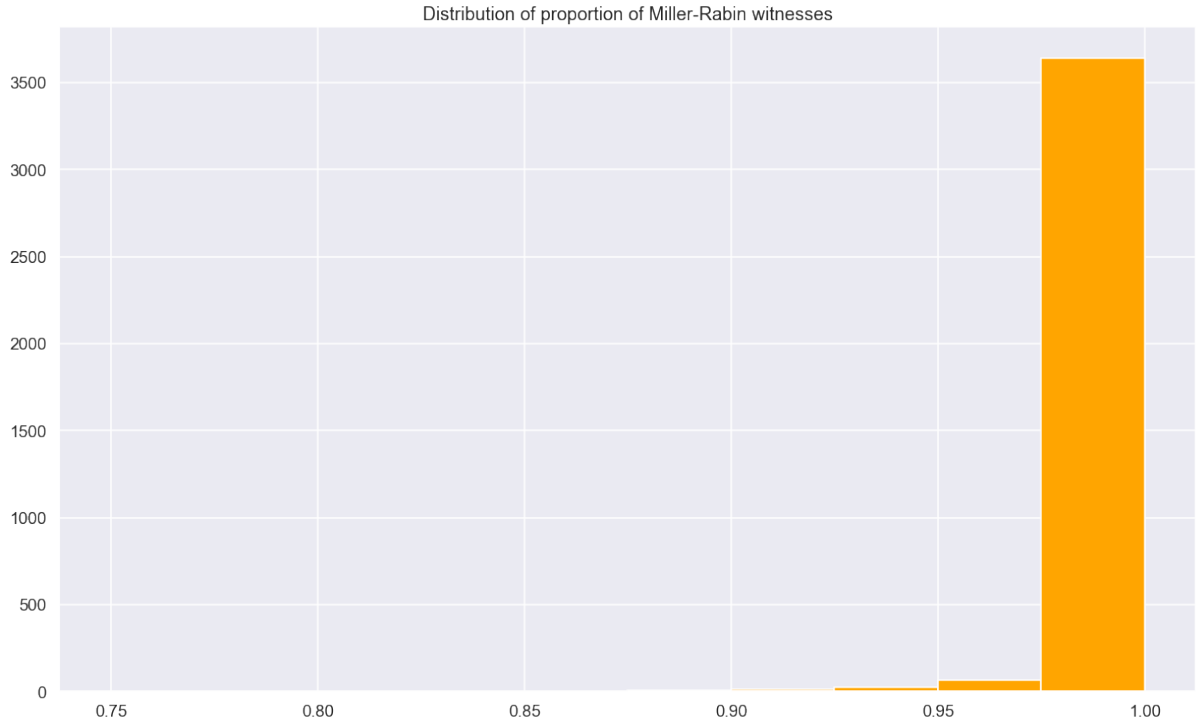
в силу того, что $-1 \not\equiv 1 \bmod n'$ (т.к. $n' \geqslant 3$). Таким образом, $a \notin G_n$, что завершает доказательство данного случая, а с ним и всей теоремы. ∎

Теорема 2.1 доказазывается аналогичным образом, оценка 1/4 на число не свидетелей достигается за счёт двукратного применения приёма с собственной подгруппой. Полное доказательство описано в [2].

3.3 Сертификат

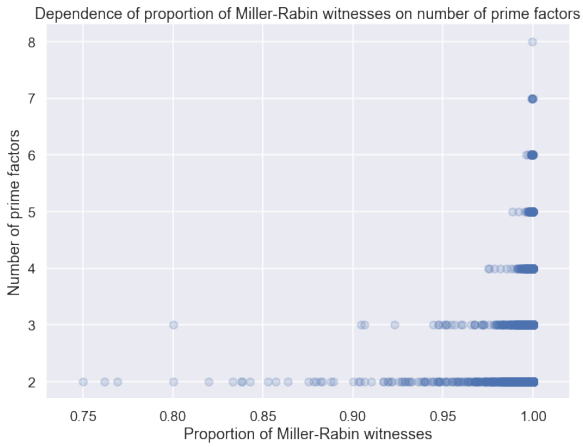
Из алгоритма видно, что сертификатом является определённый выше свидетель — число a, для которого не выполнено необходимое условие простоты. Сложность проверки сертификата составляет $O(\log^3 n)$, если умножение производится за $O(\log^2 n)$, то есть проверка сертификата полиномиальна.

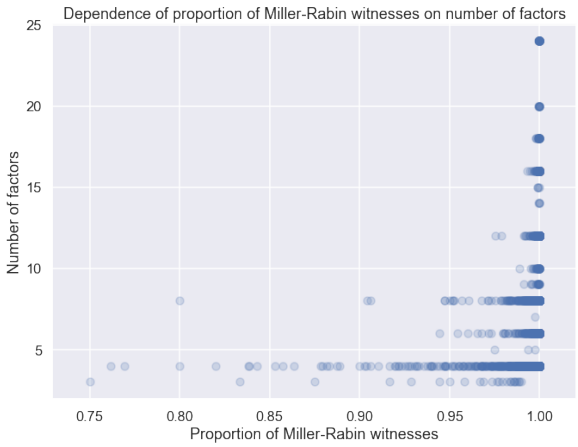
На самом деле, для большинства составных чисел доля свидетелей гораздо выше 75%. Для демонстрации этого факта построим гистограмму долей свидетелей для всех составных чисел до 10^4:



3.3.1 Зависимость числа свидетелей от строения числа

Построим графики зависимости доли свидетелей от числа простых делителей с учётом кратности и числа всех делителей числа:





С учётом коэффициентов корреляции Пирсона на уровне 0.12 в обоих случаях можно заключить, что число свидетелей зависит от количества делителей числа.

В следующей секции будет описан алгоритм из **RP**. В силу нетривиальности оного будет дано неформальное интуитивное описание. Полный алгоритм и доказательство корректности доступно в [4].

4. Алгоритм Эдельмана-Хуана

Для начала проиллюстрируем метод проверки на простоту, используемый в алгоритме Эдельмана-Хуана.

4.1 Пример

Рассмотрим следующее доказательство простоты числа 11:

(1) $(4-1, 11) = 1$

(2) $4^5 = 1024 \equiv 1 \bmod 11$

(3) 5 — простое число

Пусть 11 не является простым. Тогда существует простое $p \leqslant \sqrt{11} < 4$, делящее 11. Из (1) получаем, что $4 \bmod p \not\equiv 1$ в $Z/pZ^* \implies ord(4 \bmod p) \neq 1$. Из (2) следует, что $ord(4 \bmod p) \mid 5$. Наконец, из (3) получаем, что $ord(4 \bmod p) = 5$, что невозможно, так как $p < 4$.

Заметим, что без (3) рассуждение выше дало бы сведение доказательства простоты числа 11 к простоте числа 5. Идея сведения доказательства простоты одного числа к простоте другого является ключевой в описываемом алгоритме.

Построим алгоритм на технике из примера:

Алгоритм: Для достаточно больших p за вероятностно полиномальное время можно свести доказательство простоты p к простоте $\dfrac{p-1}{2}$. Для этого надо лишь перебирать случайные $a \in Z_{>0}$, пока не будет выполнено

(1) $(a-1, p) = 1$

(2) $a^{\frac{p-1}{2}} \equiv 1 \bmod p$

Для $p > 2$ существует $\dfrac{p-3}{2}$ чисел $a < p$, удовлетворяющих этим требованиям, так что алгоритм будет работать за вероятностно полиномиальное время.

Возвращаясь к проверке на простоту числа 11, можно было свести простоту числа 5 к простоте числа 2, которую можно проверить явно. В то же время данный алгоритм не применим в общем случае, ибо сведение простоты числа 13 к простоте числа 6 бесполезно.

В следующем подпункте будет описан рабочий алгоритм, обеспечивающий сведение.

4.2 Алгоритм Гольдвассер-Килиана

Алгоритм Гольдвассер-Килиана обходит описанную выше проблему за счёт использования групп, отличных от Z/pZ^*. Для этого введем эллиптические кривые:

Определение 4.1. Эллиптической кривой над полем $\mathbb{F}_p$ называется кривая, задаваемая уравнением вида

$$y^2 = x^3 + Ax + b \quad (a, b \in \mathbb{F}_p), \tag{1}$$

для которой выполнено $\Delta = 4A^3 + 27B^3 \neq 0$.

4.2.1 Гипотеза Римана для конечных полей

Дзета-функция Римана $\zeta(s)$ для $\mathrm{Re}(s) > 1$ определяется как

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$$

и аналитически доопределяется на всей комплексной плоскости (см. [7]). Оригинальная гипотеза Римана утверждает, что все нули дзета-функции Римана, не принадлежащие вещественной оси, лежат на прямой $\mathrm{Re}(s) = 1/2$.

Гипотеза Римана для кривых над конечными полями имеет несколько эквивалентных формулировок, приведём здесь две из них, которые в дальнейшем пригодятся при описании алгоритма. Обозначая эллиптическую кривую как E, можно определить её дзета-функцию с помощью следующей формулы:

$$Z_E(t) = \frac{1 - a_p(E)t + pt^2}{(1-t)(1-pt)},$$

где зависимость $Z_E(t)$ от E проявляется в коэффициенте $a_p = p - N_p$, где

$$N_p = \#\{\text{решений (1) в } \mathbb{F}_p\}$$

Тогда гипотеза Римана для E утверждает, что

$$Z_E(q^{-s}) = 0 \implies \mathrm{Re}(s) = 1/2$$

Теперь переформулируем гипотезу в качестве ограничения на a_p. Пусть значения $x^3 + ax + b$ равномерно распределены на $\mathbb{F}_p$ в зависимости от x. В силу того, что p нечётно, из основной теоремы о гомоморфизме половина из $p-1$ ненулевого значения $x^3 + ax + b$ не будет квадратом какого-либо элемента в $\mathbb{F}_p$, то есть не будет точкой на E. Для другой половины для некоторого $y \in \mathbb{F}_p$ будет справедливо $(\pm y)^2 = x^3 + ax + b$, что даёт две точки на E для каждого из $\dfrac{p-1}{2}$ значений x. Таким образом, ожидаемым значением N_p является $1 + 2 \cdot \dfrac{p-1}{2} = p$, и a_p получается из ожидаемого значения N_p.

Рассуждения выше приводят нас к эквивалентной формулировке гипотезы Римана для E:

$$|N_p - p| \leqslant 2\sqrt{p} \tag{2}$$

Действительно, если $Z_E(q^{-s}) = 0$, то q^s является корнем многочлена

$$f(u) = u^2 - a_p u + p$$

Заметим, что неравенство (2) истинно тогда и только тогда, когда дискриминант $f(u)$ $D = a_p^2 - 4p \leqslant 0$, а это равносильно тому, что корни u_1 и u_2 многочлена $f(u)$ либо чисто мнимые, либо равны. Это условие, в свою очередь, эквивалентно $|u_1| = |u_2|$. В силу теоремы Виета свободный член $p = u_1 u_2$, откуда оба корня $f(u)$ равны по модулю $\sqrt{p}$. Таким образом, получили утверждение

$$Z_E(p^{-s}) = 0 \implies (|p^s| = \sqrt{p} \iff \mathrm{Re}(s) = 1/2),$$

что доказывает эквивалентность (2) гипотезе Римана для кривых над конечными полями.

4.2.2 Описание алгоритма

Алгоритм: Случайным образом ищем $a, b \in Z/pZ$, удовлетворяющие следующим условиям:

(1) $f(x) = x^3 - ax + b$ не имеет кратных корней

(2) число рациональных точек на эллиптической кривой $y^2 = f(x)$ равно $2 * m$ для некоторого $m \in Z$

Далее ищем пару $s, t \in Z/pZ$, таку что

(1) точка $v = \langle s, t \rangle$ лежит на кривой

(2) $m * v = 1$ в группе, ассоциированной с кривой

Можно показать, что что если найти v, удовлетворяющую условию, то доказательство простоты p сводится к доказательству простоты m (то есть получено утверждение m простое $\implies$ p простое). Кроме того, из гипотезы Римана для кривых над конечными полей $m \approx \dfrac{p}{2}$ в силу (2).

С другой стороны, для того, чтобы описанный выше алгоритм был вероятностно полиномиальным, требуется, чтобы на случайно сгенерированной эллиптической кривой с высокой вероятностью было дважды простое число рациональных точек. Гипотеза Римана (2) гарантирует, что число точек находится в интервале

$$[p + 1 - 2\sqrt{p}, p + 1 + 2\sqrt{p}]$$

В силу того, что этот интервал слишком мал, вероятностная полиномиальность описанного выше алгоритма до сих пор не доказана. Эта проблема решена в алгоритме Эдельмана-Хуана, описанном ниже.

4.3 Алгоритм Эдельмана-Хуана

Введём понятие гиперэллиптических кривых:

Определение 4.2. Гиперэллиптической кривой рода $g > 1$ называется алгебраическая кривая, задаваемая уравнением вида

$$y^2 + h(x)y = f(x),$$

где $f(x)$ — многочлен степени $n = 2g + 1 > 4$ или с $n = 2g + 2 > 4$ различными корнями и $h(x)$ — многочлен степени $< g + 2$.

Для расширения множества чисел, для которых существует сведение, описанное в предыдущем пункте, в алгоритме Эдельмана-Хуана используется обобщённый алгоритм Гольдвассер-Килиана. Кроме того, вводится сведение нового типа: в нём эллиптические кривые заменяются на матрицы Якоби гиперэллиптических кривых второго рода. Тогда сведение нового типа вычисляется за вероятностно полиномиальное время следующим образом:

Алгоритм:

1) Случайным образом ищем многочлен $f \in Z/pZ[x]$ степени 6, который не имеет кратных корней.

2) Считаем число n рациональных точек на матрице Якоби кривой, задаваемой уравнением $y^2 = f(x)$

3) Находим такие $s, t \in Z/pZ, t \neq 0$, что точка $\langle s, t\rangle$ лежит на кривой и $v = \langle s, t\rangle - \langle s, -t\rangle$ (или, точнее, $v = \phi(\langle s, t\rangle) - \phi(\langle s, -t\rangle)$, где ϕ — вложение кривой в группу матриц Якоби) удовлетворяет равенству $n * v = 1$ в группе, ассоциированной с матрицами Якоби.

Получаем сведение доказательства простоты p к доказательству простоты n, которое, как и раньше, работает только если n — простое. В силу того, что n простое с высокой вероятностью, новый алгоритм является вероятностно полиномиальным.

С другой стороны, в силу гипотезы Римана для кривых над конечными полями $n \approx p^2$ (аналогично (2), квадрат возникает в силу того, что степень f возрасла вдвое). Таким образом, алгоритм Эдельмана-Хуана обходит числа, для которых не выходит получить сведение с помощью обобщённого алгоритма Гольдвассер-Килиана, производя несколько итераций нового сведения. В силу почти случайного выбора n данный способ позволяет избежать чисел из плохого множества.

Для завершения построения алгоритма остаётся научиться быстро считать число рациональных точек на эллиптических кривых, алгоритм Шуфа [5] справляется с этой задачей за полиномиальное время. Получаем

Теорема 4.1. Существует $c \in Z_{\geqslant 0}$ и полиномиально вычислимая всюду всюду определённая функция $F : Z^2_{\geqslant 0} \to \{0, 1\}$, такая что

1. $\forall n \in Z_{\geqslant 0}$, где n составное, и $\forall r \in Z_{\geqslant 0}$

$$F(n, r) = 0$$

2. $\forall p \in Z_{\geqslant 0}$, где p простое,

$$\frac{\#\{r : |r| \leqslant |p|^e \text{ и } F(n, r) = 1\}}{\#\{r : |r| \leqslant |p|^e\}} \geqslant \frac{1}{2},$$

то есть проверка на простоту лежит в **RP**. Учитывая, что в секции 3 было показано, что эта задача лежит в **coRP**, получаем, что она лежит в $\mathbf{ZPP} = \mathbf{RP} \cap \mathbf{coPR}$.

4.4 Сертификат

В данном случае сертификатом является последовательность чисел, к которым производится сведение доказательства простоты в процессе работы алгоритма, а для его проверки требуется лишь проверить корректность сведений и проверить полученное небольшое число на простоту делением на все числа в интервале $[2, \sqrt{n}]$, что делается за полином.

Наконец, рассмотрим наиболее эффективный на данный момент **RP** алгоритм проверки на простоту, алгоритм Аткина-Морейна.

5. Алгоритм ECPP

5.1 Описание

Алгоритм ECPP построен на идее сведений, аналогичной алгоритму Гольдвассер-Килиана, но для ускорения отказывается от использования неэффективного алгоритма Шуфа для подсчёта числа рациональных точек на эллиптической кривой. Вместо этого с помощью комплексного умножения случайно генерируется кривая, число рациональных точек на которой легко подсчитать.

Сведения в алгоритме ECPP основаны на следующей теореме:

Теорема 5.1. Пусть $N \in \mathbb{N}_+$, и эллиптическая кривая E задаётся уравнением

$$y^2 = x^3 + ax + b \bmod N$$

Рассматрим E над $\mathbb{Z}/N\mathbb{Z}$, используя обычный закон сложения и считая 0 нейтральным элементом на E.

Пусть m целое. Если существует простое число q, делящее m большее, чем $\left(\sqrt[4]{N}+1\right)^2$ и на E существует точка P, для которой выполнено

(1) $mP = 0$

(2) (m/q)P определено и не равно 0,

то N простое.

5.2 Сертификат

В силу того, что ECPP, также как и Гольдвассер-Килиан, производит цепочку сведений, он также создаёт сертификат, хоть и не является вероятностно полиномиальным

5.3 Исследование времени работы

5.3.1 Границы применимости

Для начала определим, насколько большие числа можно проверять на простоту пробным делением или доказывать их простоту с помощью ECPP. Для алгоритма Миллера-Рабина такой анализ бессмысленен в силу двух причин:

1. Он работает быстро даже на очень больших числах, например, $\sim 10^{200}$

2. Этот алгоритм работает в паре с ECPP, так что применимость ограничивается скоростью работы последнего

Будем использовать следующую схему исследования: берём в качестве пробного числа ближайшее простое больше степени 10, которое получается с помощью **nzmath.prime.nextPrime()**, после чего измеряем время работы пробного деления и ECPP. Отметим, что использование пробного деления для доказательства, что число составное, представляет собой частный случай доказательство простоты, так что данное исследование даст оценку снизу на время работу обоих алгоритмов.

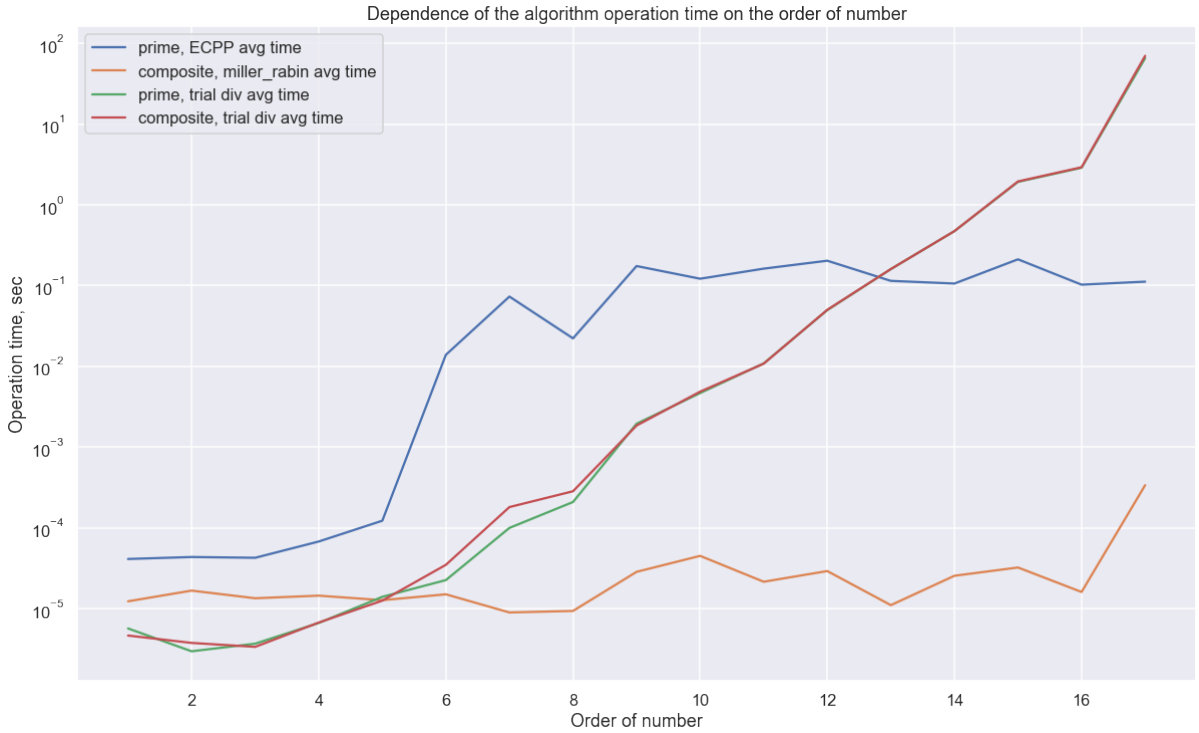
Алгоритм считается применимым для данного порядка числа, если его можно использовать для проверки на простоту некоторого массива чисел, так что ограничим эффективное время работы 10-ю секундами и построим таблицу.

$\lg n$	Время работы пробного деления, сек	Время работы ECPP, сек
5	$3.6 \cdot 10^{-5}$	10^{-4}
10	0.011	0.2
15	3.25	0.14
20	—	0.16
25	—	0.65
30	—	0.48
35	—	0.36
40	—	0.91
45	—	1.25
50	—	10.5

Таким образом, ECPP значительно расширяет диапазон чисел, с которыми можно работать за адекватное время.

5.3.2 Время работы

Теперь подробнее исследуем зависимость времени работы от размера входа, для этого разобьём отрезок $[10, 10^{17}]$ (правая граница выбрана, чтобы график строился за адекватное время) на равные интервалы в логарифмическом масштабе и будем генерировать 3 случайных числа в каждом интервале и брать следующие за ними простые, после чего усреднять время работы ECPP и пробного деления. Также на случайных составных числах с аналогичным усреднением сравним времена работы пробного деления и алгоритма Миллера-Рабина.



Из графика видно, что уже при $n \sim 10^{14}$ ECPP становится эффективнее пробного деления. Также стоит отметить, что, в отличие от пробного деления, которое требует одинаковое количество времени для доказательства простоты и факта, что число составное, Миллер-Рабин работает значительно быстрее ECPP, и уж тем более пробного деления. Более того, время его работы слабо зависит от порядка числа, что уже отмечалось ранее.

6. Практика

На практике для проверки на простоту достаточно по очереди запускать алгоритм Миллера-Рабина и ECPP, пока один из них не выдаст ответ без ошибки. Чтобы получить вероятностно полиномиальный алгоритм, можно также параллельно запустить алгоритм Эдельмана-Хуана.

7. Список литературы

[1] Д.В. Мусатов. “Сложность вычислений.”

[2] Conrad, Keith. (2017). “The Miller – Rabin Test.”

[3] Monier, Louis. (1980). “Evaluation and comparison of two efficient probabilistic primality testing algorithms.” Theoretical Computer Science. 12. 97–108.

[4] M. Adleman, Leonard & A. Huang, Ming-Deh. (1992). “Primality Testing and Abelian Varieties Over Finite Fields.” 10.1007/BFb0090185.

[5] Schoof, René. "Counting points on elliptic curves over finite fields." Journal de théorie des nombres de Bordeaux 7.1 (1995): 219-254. <http://eudml.org/doc/247664>.

[6] Atkin, A. O. L., & Morain, F. (1993). "Elliptic curves and primality proving." Mathematics of Computation, 61(203), 29–29. doi:10.1090/s0025-5718-1993-1199989-x

[7] Jasbir S. Chahal & Brian Osserman. (2008). "The Riemann Hypothesis for Elliptic Curves", The American Mathematical Monthly, 115:5, 431-442, DOI: 10.1080/00029890.2008.11920545