Содержание

Лекция 1. Односторонние функции

1 Введение

5 миров Импальяццо.

- Алгоритмика ($\mathbf{P} = \mathbf{NP}$).
- Эвристика ($\mathbf{P} \neq \mathbf{NP}$, но есть быстрый алгоритм в среднем).
- Pessiland ($\mathbf{P} \neq \mathbf{NP}$, нет ни быстрых алгоритмов, ни односторонней функции).
- Миникрипт (есть односторонние функции, но нет односторонних функций с секретом).
- Криптомания (есть односторонние функции с секретом).

В принципе, может статься еще что-то странное навроде $\mathbf{P} = \mathbf{NP}$, но на практике эти алгоритмы очень долгие или $\mathbf{P} \neq \mathbf{NP}$, но наоборот, есть какие-либо квазиполиномиальные быстрые алгоритмы.

Сначала будут обсуждаться примитивы, односторонние функции, доказательства с нулевым разглашением и прочее, потом на базе этого покажем, как построить криптографические протоколы.

Литература: конспект лекций Верещагина «Лекции по математической криптографии», черновик, Glodreich «Foundations of Cryptography», конспекты Goldwasser.

2 Односторонние функции

В криптографических задачах полиномиальность будет считаться от параметра безопасности n (неформально, длина ключа) для доказательства надёжности, и от длины шифруемого сообщения при шифровании.

Определение 1. $\{f_n\}_{n=1}^{\infty}$ — семейство односторонних функций, если:

- f регулярны по длине: $f_n : \{0,1\}^{k(n)} \to \{0,1\}^{l(n)}$.
- f вычислимы за полиномиальное время.
- f труднообратима (4 варианта: в сильном/слабом смысле, против равномерного/неравномерного обратителя).

Обратимость в слабом смысле: вероятность неудачи обратителя больше, чем некоторый обратный полином.

В сильном смысле: вероятность успеха асимптотически меньше, чем любой обратный полином.

Равномерный обратитель — полиномиальный вероятностный алгоритм.

Неравномерный обратитель — семейство схем полиномиального размера.

Задача обращения: по $f(x)$ найти $x' : f(x') = f(x)$.

Кванторная запись определения труднообратимой функции в сильном смысле: $\forall p(\cdot)\forall\{R_n\}_{n=1}^{\infty}\exists N : \forall n > N \to P_{x\sim U_k(n)}\{f(R(f(x))) = f(x)\} < \frac{1}{p(n)}$.

R в определении пробегает по всем семействам схем или вероятностным обратителям в зависимости от вида обратителя. Определение в слабом смысле отличается первым квантором.

Задача. Может ли семейство $f_n : |\mathrm{Im} f_n| = poly(n) = s(n)$ быть труднообратимом в слабом смысле?

Неравномерный обратитель: можно «зашить» в схему по одному прообразу от каждого класса.

Равномерный: берёт случайный x, вычисляет $f(x)$, если $y = f(x)$, вернуть x. Можно подобрать такое число повторений N, чтобы вероятность ошибки была мала. Идея: классы бывают большие (размера $> 2^{l(n)} \cdot \varepsilon$), и маленькие. Вероятность неуспеха для больших классов не больше $(1-\varepsilon)^N$, а для маленького класса можно оценить единицей. Тогда общая вероятность ошибки для слуйчайного x не больше $s(n) \cdot \varepsilon + (1-\varepsilon)^N$. Если ε взять как $\frac{1}{2s(n)q(n)}$, а $N = \frac{n}{\varepsilon}$, то сумма будет не больше $\frac{1}{q(n)}$ для любого полинома $q(n)$.

Задача. f — односторонняя функция. Верно ли, что $g(x) = f(x)f(x)$ тоже одностороняя? Верно ли, что $h(xy) = f(x)f(y)$ будет односторонней?

Если g односторонняя, то $\exists R_g$, которая обращает g. Тогда $R_f(y) = R_g(yy)$ обращает f.

Если R_h обращает h, то можно построить такой обратитель f: берём случайный y, считаем $f(y)$ и возвращаем первую часть $R_h(f(x)f(y))$, если всё нормально, иначе нужно повторить процедуру.

Хорошие значения x - это те, для которых доля пар (x, y) больше или равна ε. Остальных значений x мало. Аналогичными предыдущей задаче рассуждениями можно получить оценку.

Лекция 2. Слабо и сильно односторонние функции

3 Построение сильно односторонней функции из слабой

Напоминание:

Определение 1. Слабо односторонняя функция $f(x)$ — это такая, что $\exists p(x) \geqslant 0 \forall\{C_n\}_{n=1}^{\infty}\exists N\forall n \geqslant N \to P(f(C_n(f(n))) = f(x)) < 1 - \frac{1}{p(n)}$, где C_n — семейство схем полиномиального размера, а $f(x)$ вычислима за полиномиальное время.

Определение 2. Сильно односторонняя функция $f(x)$ — это такая, что $\forall p(x) \geqslant 0 \forall \{C_n\}_{n=1}^{\infty} \exists N \forall n \geqslant N \to P(f(C_n(f(n))) = f(x)) < \frac{1}{p(n)}$.

Теорема 1. *Если существует слабо односторонняя функция, то существует и сильно односторонняя.*

Доказательство. Рассмотрим функцию $F(x_1, \ldots, x_N) = (f(x_1), \ldots, f(x_N))$. Ясно, что такая функция защищена от наивных обратителей, которые пытаются обратить каждую компоненту по отдельности. Однако, неясно, почему не существует более сложного и более эффективного обратителя.

Поэтому мы возьмем гипотетический обратитель R_F для F в обратитель R_f для f. Обратитель $R_f(y) = R_F(y, f(x_2), \ldots, f(x_N)) \mid_1$ может не преуспеть, так как при фиксированной первой компоненте вероятность успеха может быть мала. Однако, мы можем запускать обратитель R_F много раз, поэтому сделаем так:

```
for (int i = 0; i < n; ++i) {
    for (int j = 0; j < K; ++j) {
        x_1, ..., x_n = gen_random();          // except i
        X = R_F(f(x_1), ..., y, ... , f(x_n)); // except i
        if (f(X) == y) {
            return X;
        }
    }
}
```

Для всех $i = 1, \ldots, n$ K раз выберем случайные $x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n$, и запустим $R_F(f(x_1), \ldots, f(x_{i-1}), y, f(x_{i+1}), \ldots, f(x_n))$ и выберем i-ю компоненту x. Если $f(x) = y$, вернем x.

Пусть $\rho_i(x) = P\{f(R_F(\ldots)) = f(x)\}$, $\rho_{\max}(x) = \max\limits_{i=1,\ldots,N} \rho_i(x)$. x бывает двух видов: такой, что $\rho_{\max}(x) \geqslant \varepsilon$ и такой, что $\rho_{\max} < \varepsilon$, доля последних равна δ.

Вероятность неудачи в таком случае $R_f \leqslant \delta + (1-\varepsilon)^k$. Если F — не сильно одностороняя, то вероятность успеха $R_F > \frac{1}{q(n)}$.

Для $(x_1, \ldots, x_n)$ вероятность, что все x_i хорошие $\leqslant (1-\delta)^N$, а если хотя бы один x плохой, то условная вероятность обращения $R_F < \varepsilon$.

Тогда вероятность успеха $\frac{1}{q(n)} < R_F < \varepsilon + (1-\delta)^N$. При $\varepsilon = \frac{1}{2q(n)}$ получается, что $(1-\delta)^N > \frac{1}{2q(n)}$. При $N = np(n) \Rightarrow \delta < \frac{1}{2p(n)}$. За счёт выбора K можем сделать $K = nq(n)$ и тогда $(1-\varepsilon)^K < \frac{1}{2p(n)}$. В итоге $\delta + (1-\varepsilon)^K < \frac{1}{p(n)}$, что означает, что f не слабо односторонняя. □

4 Примеры «односторонних» функций

Функция Рабина: $(x, y) \mapsto (x^2 \mod y, y)$. $y = p \cdot q, 0 \leqslant x < y$, притом p, q — простые числа вида $4k+3$.

Функция RSA: $(x, y, z) \mapsto (x^z \mod y, y, z)$.

$P\{f(R_n(f(x))) = f(x)\}$ определяется по всем $x \in D_n$, притом требование к области D_n таково, что нужно уметь порождать случайные элементы D_n, то есть существует полиномиальный вероятностный алгоритм, порождающий случайную величину, статистически близкую к равномерной на D_n (расстояние между любыми двумя событиями меньше любого обратного полинома).

Можно отметить, что у функции Рабина, например, есть так называемый «секрет» (разложение $y = p \cdot q$), благодаря которому можно расшифровать сообщение. Более формально определим

Определение 3. Семейство односторонних функций с секретом $\{f_\alpha\}_{\alpha \in A}$: $f_\alpha : D_\alpha \to R_\alpha$ это такие функции, что существуют 4 алгоритма:

- Генератор: $1^n \to (\alpha, \tau)$, генерирует ключ и секрет.
- Сэмплер: $\alpha \mapsto$ случайный элемент D_α (с точностью до статистической близости).
- Вычислитель: $(\alpha, x) \mapsto f_\alpha(x)$.
- Обратитель: $(\alpha, \tau, y) \mapsto f_\alpha^{-1}(y)$.

Притом $(\alpha, y) \mapsto f_\alpha^{-1}(y)$ труднообратимо в обычном смысле.

Улучшенная односторонняя перестановка с секретом: y выбирается как случайный элемент D_α, а обратитель помимо α и y получает случайные биты, использованные при порождении y (при этом они все равно ему не помогают).

5 Генераторы псевдослучайных чисел

Определение 4. G — генератор псевдослучайных чисел, если

- $G : \{0,1\}^n \to \{0,1\}^{p(n)}$.
- G вычислима за полином.
- $\forall \{D_n\}_{n=1}^{\infty} \forall q(\cdot) \to \exists N : \forall n > N \to \left| P_{x \sim U_n}(D_n(G(x)) = 1) - P_{y \sim U_{p(n)}}(D_n(y) = 1) \right| < \frac{1}{q(n)}$.

Ясно, что генератор должен быть односторонней функцией, так как иначе обратитель мог бы отличить вывод генератора от случайного вывода.

Теорема 2. *Если существует односторонняя функция, то существует и генератор.*

Мы докажем ослабленную версию этой теоремы:

Теорема 3. *Если существует односторонняя перестановка, то существует и генератор.*

Определение 5. Трудный бит. Схематически: $x \mapsto f(x), x \mapsto b(x)$ вычисляются легко, $|b(x)| = 1$. При этом по $f(x)$ сложно вычислить $b(x)$: $\forall q(\cdot)\forall\{P_n\}_{n=1}\infty\exists N\forall n > N|P(P_n(f(x)) = b(x)) - \frac{1}{2}| < \frac{1}{q(n)}$.

Схема доказательства теоремы такая:

- Односторонняя перестановка $f \mapsto$ односторонняя перестановка с трудным битом: $g(x, y) = (f(x), y), b(x, y) = x \odot y = \bigoplus_{i=1}^{n} x_i y_i$.
- Генератор $n \to n+1$: $G(x) = g(x)b(x)$.
- Генератор $n \to p(n)$: $g(g(x))b(g(x))b(x), g(g(g(x)))b(g(g(x)))b(g(x))b(x), \ldots$.

Лекция 3. Односторонняя перестановка и генератор псевдослучайных чисел

6 XOR-лемма Яо

Теорема 1. *Если существует односторонняя перестановка $p : D_n \to D_n, D_n \subset \{0,1\}^{k(n)}$, то существует генератор псевдослучайных чисел.*

Доказательство. Напоминание: схема доказательства теоремы:

- Односторонняя перестановка $f \mapsto$ односторонняя перестановка с трудным битом (декодирование списком кода Адамара и дерандомизации с помощью попарной независимости).
- Генератор $n \to n+1$: $G(x) = g(x)b(x)$ (XOR-лемма Яо).
- Генератор $n \to p(n)$: $g(g(x))b(g(x))b(x), g(g(g(x)))b(g(g(x)))b(g(x))b(x), \ldots$ (hybrid argument).

Сначала сделаем второй шаг.

Определение 1. $b(x)$ — трудный бит для $f(x)$, если он полиномиально вычислим и $\forall p(\cdot)\forall\{P_n\}_{n=1}^{\infty}\exists N : \forall n \geqslant N \to \left|P(P_n(f(x)) = b(x)) - \frac{1}{2}\right| < \frac{1}{p(n)}$.

Лемма. *$b(x)$ — трудный бит для $f(x) \Rightarrow G(x) = f(x)b(x)$ — генератор псевдослучайных чисел.*

Доказательство. Если существует отличитель для $G(x)$, то $\exists s(\cdot)\exists\{D_n\}_{n=1}^{\infty}\forall N\exists n > N : |P_x(D_n(f(x)b(x)) = 1) - P_y(D_n(y) = 1)| \geqslant \frac{1}{s(n)}$. Можно считать, что выражение под модулем положительно, так как для тех n, для которых это не так, можно инвертировать вывод D_n.

Рассмотрим варианты для $D(f(x)0) = \alpha, D(f(x)1) = \beta$.

- $\alpha = \beta \Rightarrow$ значение предсказателя случайно.
- $\alpha = 0, \beta = 1 \Rightarrow$ предсказатель возвращает 1.

- $\alpha = 1, \beta = 0 \Rightarrow$ предсказатель возвращает 0.

Обозначим A, B, C, D — события для $00, 01, 10, 11$ соответственно. $A_0, A_1 \subset AB_0, B_1 \subset B \dots$ разбиения по значениям трудного бита, $a_0, a_1, \dots$ — их вероятности.

$P(D_n(f(x)b(x)) = 1) = b_1 + c_0 + d_0 + d_1$,
$P(D_n(y) = 1) = \frac{b_0+b_1}{2} + \frac{c_0+c_1}{2} + d_0 + d_1$.
Тогда разность $\Delta = \frac{b_0+b_1}{2} + \frac{c_0+c_1}{2} \geqslant \frac{1}{s(n)}$.
Успех предсказателя: $\frac{a_0+a_1}{2} + b_1 + c_0 + \frac{d_0+d_1}{2} = \frac{1}{2} + \Delta \geqslant \frac{1}{2} + \frac{1}{s(n)}$. $\square$

Почему XOR-лемма? Потому что $P(f(x)) = D(f(x)r) \oplus r \oplus 1$.

7 Построение генератора любой длины

Теперь сделаем генератор $n \to q(n)$. Для начала рассмотрим $G(x) = f(f(x))b(f(x))b(x)$, что должно быть вычислительно неотличимо от xr_1r_2.

$xr_1r_2 \sim f(x)r_1r_2$, так как x и $f(x)$ одинаково распределены (так как f — перестановка). $f(x)r_1r_2 \sim f(x)b(x)r_2$ по определению G. Далее, $xr_2 \sim f(x)b(x) \Rightarrow xr_1r_2 \sim f(f(x))b(f(x))b(x)$.

Для любого константного увеличения можно сделать точно также. Для $n \to q(n)$ делаем так:

$$h_0(x) = xr_1r_2 \dots r_{q(n)}$$

$$\vdots$$

$$h_{q(n)}(x) = f^{q(n)}(x)b(f^{q(n)-1}(x)) \dots b(x)$$

Хотим доказать, что $h_{q(n)} \sim h_0(x)$. Если $P(D_n(h_{q(n)}(x)) = 1) - P(D_n(h_0(x)) = 1) \geqslant \frac{1}{s(n)}$, то $\exists m : P(D_n(h_m(x)) = 1) - P(D_n(h_{m-1}(x)) = 1) \geqslant \frac{1}{s(n)q(n)}$, что невозможно аналогично пункту $n \to n+2$.

Теорема 2 (Левин-Голдрайх)**.** *Пусть f — односторонняя перестановка, то $g(xy) = f(x)y$ тоже одностороняя перестановка, а $b(xy) = x \odot y = \bigoplus_{i=1}^{n} x_i y_i$ — трудный бит для g.*

Доказательство. Первая часть очевидна, если f — односторонняя пересатновка, то и g тоже перестановка, легко вычисляется и если g можно обратить, то обратить можно и f. Для доказательства второй части воспользуемся кодом Адамара.

Код Адамара: $x \mapsto (x \odot z)_{z \in \{0,1\}^n}$ слово длины n превращает в слово длины 2^n. Его можно воспринимать как значение всех линейных функций на входе x или как значение на всех входах линейной функции, заданной x.

Пусть $\hat{f}(z)$ совпадает с $f(z)$ на доле входов z равной $\frac{3}{4} + \varepsilon$. Тогда можно восстановить $f(z) = \hat{f}(z+r) + \hat{f}(r)$ и с вероятностью $> \frac{1}{2}$ мы восстановим $f(z)$. Повторив много раз, можем узнать $f(e_i) = x_i$.

Для доли повреждения $\frac{1}{2}$ декодировать уже не получится, но можно декодировать списком: имея доступ к $\hat{f}(z)$ как к оракулу, напечатать полиномиальный список в котором с вероятностью $\geqslant \frac{1}{2}$ находится вектор x, определяющий f.

Задача. В шаре с центром в любой точке и радиусом (в смысле расстояния Хемминга) $\frac{1}{2} - \varepsilon$ находится $poly\left(\frac{1}{\varepsilon}\right)$ кодовых слов.

Запишем равенство: $f(z) = \hat{f}(z + r) + f(r)$, которое должо быть выполнено в $\geqslant \frac{1}{2}$ случаев. Непонятно только, откуда взять $f(r)$.

Идея попарной независимости: проведем процедуру выше для некоторого числа попарно независимых случайных r. Возьмем случайные независимые в совокупности вектора $u_1, \ldots, u_l$ и вектора $r_1, \ldots, r_{2^l-1}$ построим как $r_a = a_1u_1 + \ldots + a_lu_l$. Тогда $r_1, \ldots, r_{2^l-1}$ попарно независимы. Алгоритм будет следующий:

```
u_1, ..., u_l := random()
for (f(u_1), ... f(u_l) in {{0,1}^n}^l) {
    for (int a = 1; a < 2^l - 1; ++a) {
        f(r_a) = a_1 f(u_1) + ... + a_l f(u_l)  // linearity
        f(e_i) = f_hat(e_i + r_a) + f(r_a)
    }
    choose f(e_i) as majority for all a
    add f(e_1), ..., f(e_l) in list
}
```

Утвержается, что по неравенству Чебышёва при большом числе повторений с вероятностью больше, чем $\frac{1}{2}$ декодирование произведено верно.

Теперь, если g — это односторонняя функция, $h(xy) = g(x)y, b(xy) = x \odot y = f(y)$ и есть предсказатель b, то можно с его помощью построить $\hat{f}(y)$, совпадающую на доле $\frac{1}{2} + \varepsilon$, что можно декадировать списком $x_1, \ldots, x_m$ и каждый x проверить непосредственно.

Несмотря на то, что $\hat{f}$ экспоненциально длинная, но нам нужно только занчение в полиноме точек, которые мы и запомним (или можно относиться к $\hat{f}$ как к оракулу).

□

□

Лекция 4.

8 Псевдолучайные функции с неадаптивным отличителем

Определение 1. Семейство функций $\{f_s^n\} : f_s^n : \{0,1\}^n \to \{0,1\}^n, s \in \{0,1\}^{p(n)}$ называется псевдослучайным, если:

- Существует полиномиальный алгоритм, который по s и x вычисляет $f_s^n(x)$.

- Надёжность против неадаптивных отличителей:
$\forall q(\cdot)\ \forall\{D_n\}\forall\{x_1, \ldots, x_{q(n)}\}\ \forall w(\cdot)\ \exists N : \forall n > N \rightarrow$
$|P_s(D_n(x_1, \ldots, x_{q(n)}, f_s(x_1), \ldots, f_s(x_{q(n)})) = 1) -$
$P_g(D_n(x_1, \ldots, x_{q(n)}, g(x_1), \ldots, g(x_{q(n)})) = 1)| < \frac{1}{w(n)}$

Теорема 1. *Если существует генератор псевдослучайных чисел из* $\{0,1\}^n$ *в* $\{0,1\}^{2n}$*, то существует и семейство псевдослучайных функций.*

Доказательство. Конструкция такова: для некотрого x длины n делаем следующее:

- Считаем $G(s) = s_0 s_1$, если 1й бит x равен 1, то берем s_1, иначе s_0.

- Считаем $G(s_{x_1}) = s_{x_1 0} s_{x_1 1}$, выбираем одну из половин в зависимости от x_2.

- Продолжаем аналогично.

Доказательство индукцией по дереву: нарисуем бинарное дерево, которое является частью полного бинарного, содержащей $x_1, \ldots, x_{q(n)}$. На каждом следующем уровне мы имеем $s_{a_1}, \ldots, s_{a_r}$, однако в силу псевдослучайности мы можем вычислительно неотличимо заменить их на действительно случайные значения. Поскольку размер дерева полиномиален, то мы использовали вычислительную неотличимость полиномиально много раз, что делать можно.

Более формально: если $G(y) = G_0(y)G_1(y)$, то можно записать $f_s(x) = G_{x_n}(G_{x_{n-1}}(\ldots Gx_1(s)\ldots))$.
$h_{i,t}(x) = G_{x_n}(G_{x_{n-1}}(\ldots Gx_{i+1}(t_{x_1 \ldots x_i})\ldots))$, $|t| = 2^{n+i}$.
$h_{0,t}(x) = f_t(x)$, $h_{n,t}(x)$ — случайная функция. Цепочка эквивалентностей приводит к тому, что они вычислительно неотличимы. □

9 Адаптивные отличители

Пример 1. Пример, когда адапитивный отличитель сильнее неадаптивного: пусть есть f, такая что:
$f(0\ldots0) = v$ — случайное, $f(v) = 0\ldots0$, все остальные слова случайны.

Адаптивный отличитель легко справится с такой задачей, а для неадаптивного отличителя вероятность найти нужное значение v очень мала.

Мы воспользуемся тем, что адаптивные алгоритмы — это то же самое, что алгоритмы с подсказкой и доступом к оракулу-функции. Алгоритм получает на вход 1^n и подсказку a_n длины $poly(n)$.

$A^g(1^n, a_n)$ — это результат работы такого алгоритма с функцией g в качестве оракула.

Определение 2. Систему псевдослучайных функций будем называть усточивой относительно адаптивного отличителя, если $\forall A$ — отличителя $\forall q(\cdot)\, \exists N$: $\forall n > N \to |P_s(A^{f_s}(1^n, a_n) = 1) - P_g(A^g(1^n, a_n) = 1)| < \frac{1}{q(n)}$.

Утверждение 1. Построенная система функций устойчива относительно адаптивного отличителя.

Доказательство. Доказательство в целом такое же, только одного общего дерева нет, оно стоится по ходу алгоритма. Однако в ходе рассуждений ничего особо не меняется. □

Вариации с параметрами могут быть следующие:

- Уменьшение длины - легко, если уменьшить длину выхода, ничего не нарушится.
- $f_s : \{0,1\}^* \to \{0,1\}^{r(n)}, s \in \{0,1\}^{p(n)}$. Используется генератор G : $\{0,1\}^{p(n)} \to \{0,1\}^{2p(n)+r(n)}$, $G(s) = \underbrace{G_0(s)}_{p(n)}\underbrace{G_1(s)}_{p(n)}\underbrace{G_2(s)}_{r(n)}$, а функции вычисляются так: $f_s(x) = G_2(G_{x_k}(G_{x_{k-1}}(\ldots)))$.

Лекция 5. Шифрование с открытым и закрытым ключом

10 Принципиальная схема шифрования

Шифрование с закрытым ключом: есть $Encoder(m, d)$, который передает сообщение c полиномиальной длины $Decoder(d, c) \to m$. Нужно чтобы перехватчик $A(c)$ не мог восстановить m.

Шифрование с открытым ключом: $Encoder(m, e)$ передает c программе $Decoder(c, d) \to m$. Ключи e, d у них разные, и перехватичик $A(c, d)$ может пользоваться одним из них.

Для закрытого ключа есть идеальная, но довольно бесполезная процедура: передать $m \oplus d$, где d — случайная строка. Есть две проблемы: ключ по длине равен сообщению (если мы можем обменяться такими ключами, то почему не можем обменяться сообщениями?), но даже если предположить, что мы заранее договорились о закрытом ключе, то остается проблема того, что шифр одноразовый: если известно $m_1 \oplus d$ и $m_2 \oplus d$, то можно узнать $m_1 \oplus m_2$, что может быть полезной информацией.