Лекция 8

Теорема Тодá (первая часть)

(Конспект: О. Сергеева)

8.1 Класс PP

Определение 8.1. Язык L принадлежит классу языков **PP**, если существует полиномиальная по времени НМТ M такая, что

$$x \in L \Leftrightarrow P\{M(x) = 1\} > \frac{1}{2}.$$

(Т.е. в дереве её вычислений более половины листьев соответствуют принимающим вычислениям.)

Заметим, что $\mathbf{NP} \subseteq \mathbf{PP}$. Действительно, пусть $L \in \mathbf{NP}$, M — полиномиальная по времени НМТ, которая его принимает. Построим по ней НМТ N, которая принимает слово из L с вероятностью $> 1/2$, слово не из L — с вероятностью $1/2$. Пусть x — какое-то слово. Из корня дерева вычислений N будут исходить две ветви; в одну из них мы подставим дерево вычислений M, во вторую — его же, но сделав в нём все вычисления принимающими (во все листья поставим единички). Если слово x принадлежало L, в M было хотя бы одно принимающее вычисление (хотя бы одна единичка в листьях) — т.е. в N единичек больше половины. Если x не лежало в L, в N ровно половина вычислений — принимающие.

Определение 8.2. Задача `MAJ-SAT`: дана булева формула (не обязательно в КНФ); требуется выяснить, верно ли, что количество наборов значений переменных, делающих её истинной, составляет $> \frac{1}{2}$ от общего количества наборов.

Утверждение 8.1. `MAJ-SAT` — **PP**-*полная задача.*

Доказательство. `MAJ-SAT` $\subseteq$ **PP**. Действительно, возьмём НМТ, проверяющую, что на ленте подсказки — выполняющий набор нашей формулы. Среди всех возможных подсказок «хороших» — больше половины как раз если выполняющих наборов больше половины.

С другой стороны, свести к `MAJ-SAT` любую задачу из **PP** можно, записав соответствующую НМТ при помощи булевой формулы как это делалось в теореме Кука-Левина. □

8.2 Теорема Тода́, первая часть: $\mathbf{PH} \subseteq \mathbf{BPP}^{\oplus\mathbf{P}}$

Теорема 8.1 (S. Toda). $\mathbf{PH} \subseteq \mathbf{P}^{\mathbf{PP}}$.

Эта теорема будет доказана в два этапа. Для начала нам потребуется ещё одно определение.

Определение 8.3. Язык L принадлежит $\oplus\mathbf{P}$ (parity P) $\Leftrightarrow$ существует полиномиальная по времени НМТ M, такая, что

$$x \in L \Leftrightarrow \text{количество принимающих вычислений } M \text{ на } x \text{ нечётно.}$$

(Или, что эквивалентно, существует полиномиально проверяемое отношение R, такое, что $\#\{y : R(x,y) = 1\} \not\vdots\, 2$.)

Легко доказать, что полным языком для $\oplus\mathbf{P}$ является задача $\oplus$`SAT`: верно ли, что у данной булевой формулы нечётное число выполняющих наборов.

Первым этапом доказательства теоремы Тода́ будет доказательство того, что $\mathbf{PH} \subseteq \mathbf{BPP}^{\oplus\mathbf{P}}$.

Заметим, что $\mathbf{NP} \subseteq \mathbf{BPP}^{\oplus\mathbf{P}}$. Это — ослабление утверждения $\mathbf{NP} \subseteq \mathbf{RP}^{\oplus\mathbf{P}}$, которое сразу получается из леммы Вэлианта-Вазирани: в качестве оракульного языка возьмём $\oplus$`SAT`. Нам важно различить случаи, когда у формулы ни одного и когда — один выполняющий набор, остальное не важно, и с этим наш оракул справляется.

Релятивизуем это утверждение.

Лемма 8.1. *Для любого оракула A справедливо* $\mathbf{NP}^A \subseteq \mathbf{BPP}^{\oplus\mathbf{P}^A}$.

Доказательство. Пусть $L \in \mathbf{NP}^A$, M — НМТ, принимающая L с оракулом A. Построим по M машину M', получающую на вход тройку чисел (x, r, p), такую, что если ей подать случайные r и p, то для любого x с вероятностью $> 1/\mathrm{poly}(n)$ у неё на (x, r, p) будет нечётное количество (точнее, ровно одно) принимающих вычислений, а если x будет не из L, то ни одного.

Распределение r и p будет как в лемме Вэлианта-Вазирани: выберем случайное $i \in [0\,..\,n]$, где n — длина ветви M (здесь мы пользуемся тем, что все ветви можно сделать одинаковыми по длине). Далее выберем $r \in [0\,..\,4 \cdot 2^i \cdot n^2]$ и $p \in [1\,..\,4 \cdot 2^i \cdot n^2]$ также в соответствии с равномерным распределением.

Машина M' читает x и работает на этом входе так же, как M, но в тех случаях, когда M попадает в принимающее состояние, M' проверяет, что $a \bmod p = r$ (a — строка подсказки), и выдаёт результат этой проверки. Велика вероятность того, что из всех принимающих вычислений останется ровно одно. Доказательство — такое же, как в лемме Вэлианта-Вазирани. □

Для доказательства того, что $\mathbf{PH} \subseteq \mathbf{BPP}^{\oplus\mathbf{P}}$, достаточно доказать, что $\forall i \in \mathbb{N}\ \ \Sigma^i\mathbf{P} \subseteq \mathbf{BPP}^{\oplus\mathbf{P}}$. Докажем это по индукции по i. База у нас уже доказана (лемма Вэлианта-Вазирани или лемма 8.1).

Для доказательства перехода нам потребуются три леммы.

Лемма 8.2. $\oplus\mathbf{P}^{\mathbf{BPP}^A} \subseteq \mathbf{BPP}^{\oplus\mathbf{P}^A}$.

Лемма 8.3. $\oplus\mathbf{P}^{\oplus\mathbf{P}} \subseteq \oplus\mathbf{P}$.

Лемма 8.4. $\mathbf{BPP}^{\mathbf{BPP}^A} \subseteq \mathbf{BPP}^A$.

Доказав их, получим:

$$\begin{array}{rcl}
\Sigma^{k+1}\mathbf{P} & = & \\
\mathbf{NP}^{\Sigma^k\mathbf{P}} & \subseteq & \text{(по предположению индукции)} \\
\mathbf{NP}^{\mathbf{BPP}^{\oplus\mathbf{P}}} & \subseteq & \text{(по лемме 8.1)} \\
\mathbf{BPP}^{\oplus\mathbf{P}^{\mathbf{BPP}^{\oplus\mathbf{P}}}} & \subseteq & \text{(по лемме 8.2)} \\
\mathbf{BPP}^{\mathbf{BPP}^{\oplus\mathbf{P}^{\oplus\mathbf{P}}}} & \subseteq & \text{(по лемме 8.3)} \\
\mathbf{BPP}^{\mathbf{BPP}^{\oplus\mathbf{P}}} & \subseteq & \text{(по лемме 8.4)} \\
\mathbf{BPP}^{\oplus\mathbf{P}}. & &
\end{array}$$

что и завершит доказательство первой части теоремы Тодá.

Доказательство леммы 8.4.

Замечание 8.1. Было доказано, что для любого языка из **BPP** можно выбрать вероятностную машину, принимающую этот язык, со сколь угодно малой вероятностью ошибки $2^{-\mathrm{poly}(n)}$. Поэтому можно считать, что в деревьях вычислений, которые мы будем рассматривать, все листья, кроме экспоненциально малой части 2^{-p_i} (p_i — некоторый полином от длины входа) соответствуют принимающим вычислениям; полиномы подберем, когда нам будет удобно.

Пусть $L \in \mathbf{BPP}^A$, и вероятностная машина M (с двусторонней ограниченной вероятностью ошибки $2^{-e(n)}$) обращается к L как к оракулу — можно добиться того, чтобы длина всех её веток была одинаковой и во всех ветках было одно и то же число $l(n)$ обращений к L (просто добавим «пустые вычисления» и «бессмысленные обращения» к L там, где их «не хватает»).

Вместо каждого обращения к оракулу, подставим в (дерево вычислений) M дерево вычислений соответствующей вероятностной машины N (вероятность ошибки которой — $2^{-i(n)}$). После этого в полученном дереве вычислений останутся только обращения к оракулу A.

Принимающие (соответственно — отвергающие) ветви, в которых оракул каждый раз отвечал так же, как и соответствующее (ветви) вычисление N, останутся принимающими (соответственно — отвергающими). Среди них доля ошибочных вычислений составляет не более $2^{-e(n)}$.

Сколько имеется ветвей, которые могли изменить свой статус по сравнению со статусом ветви машины M, из которой они получились? Очевидно, для каждой ветви машины M их доля составляет не более $1 - (1 - 2^{-i(n)})^{l(n)}$.

Итого доля ошибочных ветвей — не более $2^{-e(n)} + 1 - (1 - 2^{-i(n)})^{l(n)}$. Ясно, что можно подобрать $e(n)$ и $i(n)$ так, чтобы эта доля была больше $\frac{3}{4}$: например, $e(n) = n$, $i(n) = nl(n)$ (заметим, что тем самым i зависит от e, т.е. i надо выбирать после e). □

Доказательство леммы 8.2. L принадлежит левой части — значит, есть полиномиальная по времени НМТ M с оракулом $B^A \in \mathbf{BPP}^A$, принимающая каждое слово $x \in \{0,1\}^n$ из L для нечётного числа подсказок $y \in \{0,1\}^{\gamma(n)}$. Можно рассмотреть соответствующее полиномиально проверяемое с оракулом B^A отношение R:

$$R(x, y) = 1 \Leftrightarrow M \text{ принимает } x \text{ с подсказкой } y.$$

Тогда

$$x \in L \Leftrightarrow \#\{y : (x, y) \in R\} \not\vdots 2.$$

Заметим, что $R \in \mathbf{P}^{B^A} \subseteq \mathbf{P}^{\mathbf{BPP}^A} \subseteq \mathbf{BPP}^{\mathbf{BPP}^A} \subseteq \mathbf{BPP}^A$ (по лемме 8.4), т.е. существует полиномиальная по времени оракульная НМТ П, которая с оракулом A на доле $\geq 1-2^{-\pi(n)}$ (полином π выберем позднее) допустимых подсказкок $z \in \{0,1\}^{\zeta(n)}$ правильно вычисляет $R(x,y)$. Имеем:

$$L = \left\{ x \;\middle|\; \#\left\{ y : \#\{z : (x,y,z) \in L(\Pi^A)\} \geq (1-2^{-\pi(n)})2^{\zeta(n)} \right\} \not\vdots\, 2 \right\} \quad (8.1)$$

Остается доказать, что

$$L = \left\{ x \;\middle|\; \#\left\{ z : \#\{y : (x,y,z) \in L(\Pi^A)\} \not\vdots\, 2 \right\} \geq \frac{3}{4}2^{\zeta(n)} \right\} \quad (8.2)$$

(тогда этот язык, очевидно, можно распознать в $\mathbf{BPP}^{\oplus\mathbf{P}^A}$).

При фиксированном x построим таблицу (строчки занумерованы y-ми, столбцы — z-ми), в клетке (y,z) отметим результат работы Π^A для данных y, z.

Строчек, в которых много (более $(1-2^{-\pi(n)})2^{\zeta(n)}$) единиц — нечётное число для $x \in L$ и чётное для $x \notin L$; столбцов, каждый из которых содержит единицу на пересечении с каждой этих строчек (и есть надежда, что тем самым количество единиц в нём будет нужной четности), — по крайней мере $2^{\zeta(n)} - 2^{-\pi(n)}2^{\zeta(n)}2^{\gamma(n)}$.

В остальных строчках единиц очень мало (менее $2^{-\pi(n)}2^{\zeta(n)}$), поэтому они все вместе влияют на чётность не более $2^{-\pi(n)}2^{\zeta(n)}2^{\gamma(n)}$ столбцов. Итого, нужной чётностью обладают по крайней мере $2^{\zeta(n)} - 2^{-\pi(n)}2^{\zeta(n)}2^{\gamma(n)} - 2^{-\pi(n)}2^{\zeta(n)}2^{\gamma(n)} \geq \frac{3}{4}2^{\zeta(n)}$ столбцов (достаточно выбрать $\pi(n) \geq \gamma(n)+3$). □

Доказательство леммы 8.3. Пусть язык L принадлежит левой части, т.е. $x \in L \Leftrightarrow$ количество принимающих вычислений некоторой полиномиальной по времени НМТ N с оракулом $V \in \oplus\mathbf{P}$ — нечётно. Сконструируем по N и V НМТ M (без оракула), количество принимающих ветвей которой — той же чётности, что и у N^V.

Для этого в тех вершинах дерева вычислений N, где есть обращение к оракулу, вставим разветвление: одна ветвь будет соответствовать положительному ответу оракула V, и мы подставим в неё дерево вычислений[1] V, а вторая — отрицательному, мы подставим в неё дерево вычислений машины $\overline{V}$ (принимающей те и только те слова, которые V отвергает).

Лемма 8.5. *Такая $\overline{V}$ существует, т.е.* $\oplus\mathbf{P} = \mathbf{co}\text{-}\oplus\mathbf{P}$.

[1]Сейчас мы сконструируем другую ветвь по лемме 8.5; после этого следует искусственно удлинить вычисления V, чтобы они были такой же длины, как и у машины, сконструированной по лемме.

Доказательство. Вставим дополнительное разветвление (недетерминированный выбор) на первом шаге, и в левом поддереве проделаем те же вычисления, что и V, а в правом — фиктивные вычисления той же длины, из которых принимающим будет только первое. Таким образом, чётность количества принимающих вычислений сменилась на противоположную относительно V. □

Разветвление соответствует тому, что вместо того, чтобы обратиться к оракулу V, мы недетерминированно угадываем его ответ и продолжаем работу с этим ответом. Т.е. листья подставленных вычислений V или $\overline{V}$, в которых у V и $\overline{V}$ были нули, мы оставляем листьями с нулями, а в единичных листьях этих деревьев продолжаем вычисления, используя соответствующий бит в качестве ответа оракула.

Покажем, что чётность числа единиц в листьях поддерева, следующего за обращением к оракулу, после такого преобразования дерева не меняется. Ветвь, в которой мы не угадали ответ оракула, на чётность числа единиц в листьях не влияет: подставленное в эту ветвь дерево V или $\overline{V}$ имеет чётное число единиц в листьях. Теперь рассмотрим ветвь, в которой мы дали верный ответ. Поддерево, которое мы здесь подставили в листья-единицы, совпадает с тем поддеревом вычислений, которое было в исходной машине после обращения к оракулу; повторено оно нечётное количество раз, поскольку именно столько вычислений V или $\overline{V}$ закончилось листом-единицей. □