Московский физико-технический институт
Физтех-школа прикладной математики и информатики

СЛОЖНОСТЬ ВЫЧИСЛЕНИЙ

IV СЕМЕСТР

Лектор: *Мусатов Даниил Владимирович*

$h\nu$

Автор: *Киселев Николай*
Репозиторий на Github

весна 2025

Содержание

1 Класс P

1.1 Базовые определения

Будем рассматривать задачи на распознавание, т.е. дан $A \subset \{0,1\}^*$ и требуется по $x \in \{0,1\}^* \mapsto \begin{cases} 1, x \in A \\ 0, x \notin A \end{cases}$. Пусть M решает данную задачу, т.е. $\forall x (x \in A \Leftrightarrow M(x) = 1)$.

Определение 1.1. $time_M(x)$ — число шагов $M(x)$ при вычислении ответа.

Определение 1.2. $time_M(n) = \max_{x:|x|=n} time_M(x)$

Определение 1.3. $time_M(n) = O(f(n))$, если $\exists C : \forall n : time_M(n) \leqslant C \cdot f(n)$

Возникает вопрос: можно ли сказать, что $time_A(n) = \min_{M:M \text{ решает } A} time_M(n)$? Нет, но показать это достаточно сложно (Теорема Блюма).

Поэтому мы приходим к данному определению:

Определение 1.4. $\mathbf{DTIME}(t(n)) = \{A | \exists M : M(x) = 1 \Leftrightarrow x \in A, time_M(n) = O(t(n))\}$

Заметим, что для определения **DTIME**, необходимо задать модель вычислений. Обычно такой моделью выбирают многоленточную машину Тьюринга.

Определение 1.5. $\mathbf{P} = \mathbf{DTIME}(poly(n)) = \bigcup_{k=1}^{\infty} \mathbf{DTIME}(n^k)$

Тезис Черча-Тьюринга в сильной форме: Любая задача, эффективно решаемая физическим устройстом, решается за полиномиальное время на машине тьюринга.

Пример (Нетривиальные примеры задач из **P**).

1. $\mathbb{P}$ — множество простых чисел
2. Линейное программирование — как пример, нахождения максимума функции на многограннике. Эта задача не бинарная, но вот задача "достижима ли это число на многограннике" принадлежит классу **P**.
3. Симплекс-метод — алгоритм решения

1.2 Неконструктивные оценки P

Рассмотрим, например, задачу определения графа на планарность. Для этого существует два критерия: критерий Понтрягина-Куратовского: граф планарен $\Leftrightarrow$ в нем нет подграфов, гомеоморфных $K_5, K_{3,3}$. Также, существует критерий Вагнера: граф планарен $\Leftrightarrow$ в нем нет миноров $K_5, K_{3,3}$ (минор — граф, полученный из исходного удалением и стягиванием ребер). Рассмотрим свойства, которые сохраняются при удалении и стягивании ребер.

Теорема 1.1 (Робертсона-Сеймура).

1. Для любого свойства, аналогичного планарности выполнен аналог критерия Вагнера с конечным числом запрещенных миноров.

2. Наличие такого минора проверяется за полиномиальное время

Следствие. Любое такое свойство лежит в классе $\mathbb{P}$.

Но проблема в том, что мы не знаем миноров, которые необходимо проверить, чтобы найти проверить выполнение данного свойства.

1.3 Другие классы задач

Определение 1.6. $\mathbf{QP} = \mathbf{DTIME}(2^{poly(\log(n))}) = \bigcup_{c=1}^{\infty} \mathbf{DTIME}(2^{(\log n)^c})$

Определение 1.7. $\mathbf{E} = \mathbf{DTIME}(2^{O(n)}) = \bigcup_{c=1}^{\infty} \mathbf{DTIME}(2^{cn})$

Определение 1.8. $\mathbf{EXP} = \mathbf{DTIME}(2^{poly(n)}) = \bigcup_{c=1}^{\infty} \mathbf{DTIME}(2^{n^c})$

Определение 1.9. $\mathbf{EE} = \mathbf{DTIME}(2^{2^{cn}}) = \bigcup_{c=1}^{\infty} \mathbf{DTIME}(2^{2^{cn}})$

1.4 Асимптотики различных задач

Пример. $\mathsf{LOG-CLIQUE} = \{G|\omega(G) \geqslant \log_2 n\} \in \mathbf{QP}$. Является квазиполиномиальной, т.к. $C_n^{\log n} \leqslant n^{\log n}$, т.е. полный перебор осуществляется за квазиполином

Пример (Задача о доминирующем множестве в турнире)**.** непон

Пример. $\mathsf{GI} = \{(G_1, G_2) : G_1 \cong G_2\} \in \mathbf{QP}$

Пример. $\mathsf{3COL} = \{G : \chi(G) \leqslant 3\} \in E$

Теорема 1.2 (об иерархии по времени)**.** *Если* $f \ll g \Rightarrow \mathbf{DTIME}(\mathbf{f}(\mathbf{n})) \subsetneq \mathbf{DTIME}(\mathbf{g}(\mathbf{n}))$

2 Класс NP

Сейчас будем рассматривать модель вычислений — *недетерминированную машину Тьюринга* или НМТ. В отличие от обычной машины Тьюринга, функция перехода теперь многозначна (по аналогии с ДКА и НКА).

Соответственно, время работы такой машины Тьюринга — $time_M(x) = \max$ #шагов по всем вариантам перехода.

Замечание. Можем считать, что дерево переходов двоичное. Действительно, размер ветвления ограничено мощностью $|\Sigma| \cdot |Q| \cdot |\{N, R, L\}|$ — некоторая константа, не зависищая от входа. Тогда каждое m-ветвление можно заменить $\log_2 m$ 2-ветвлениями.

Ответ данной машины будем понимать следующее:

$$M(x) = \begin{cases} 1, \text{существует принимающая ветка} \\ 0, \text{иначе} \end{cases}$$

Замечание. Ответ вычисляется как дизъюнкция по всем результатам работы машины

Определение 2.1. $\mathbf{NTIME}(t(n))$ — класс языков, распознаваемых на НМТ за $O(t(n))$ шагов

Определение 2.2. $\mathbf{NP} = \bigcup_{c=1}^{\infty} \mathbf{NTIME}(n^c)$

Определение 2.3. $\mathbf{NE} = \bigcup_{c=1}^{\infty} \mathbf{NTIME}(2^{cn})$

Определение 2.4. $\mathbf{NEXP} = \bigcup_{c=1}^{\infty} \mathbf{NTIME}(2^{n^c})$

Замечание. $\mathbf{NTIME}(\mathbf{t}(\mathbf{n})) \subset \mathbf{DTIME}(\mathbf{2^{t(n)}})$

Замечание. $\mathbf{NP} \subset \mathbf{EXP}$ — за время **EXP** можно построить все дерево и вычислить ответ по определению.

2.1 Определение через сертификат

Теорема 2.1. $A \in \mathbf{NP} \Leftrightarrow \exists V(x,s)$ — *ДМТ, т.ч.* $x \in A \Leftrightarrow \exists s : V(x,s) = 1$ *и* $V(x,s)$ *работает за* $poly(|x|)$

Доказательство.

$\Leftarrow$ Расмотрим следующую НМТ, которая сначала печатает все возможные варианты s (достаточно написать полиномиальное количество символов, т.к. больше V не сможет прочесть), а потом на входе x, s запускает V. Таким образом, Получили машину M, которая в какой-то ветке напечатает нужный сертификат s и выведет $V(x,s) = 1$.

$\Rightarrow$ Возьмем в качестве сертификата код нужной ветви в машине M (0, если надо идти вправо, 1, если влево). Оно и будет нашим сертификатом s. Машина V будет спускаться, в соответствии с сертификатом, по дереву переходов. Тогда сертификат существует $\Leftrightarrow$ сущетсвует принимающая ветвь $\Leftrightarrow A \in \mathbf{NP}$.

□

Упражнение. *Сформулировать и доказать аналогичную теорему для классов* $\mathbf{NE}, \mathbf{NEXP}$

Утверждение 2.1. *1.* $A, B \in \mathbf{P} \Rightarrow A \cap B, A \cup B, \overline{A} \in \mathbf{P}$

2. $A, B \in \mathbf{NP} \Rightarrow A \cap B, A \cup B \in \mathbf{NP}$

Замечание. Вообще говоря, $\overline{A} \in \mathbf{NP}$ — открытый вопрос. Нельзя просто инвертировать значение машины M (пусть мы получим машину $\overline{M}$): тогда $x \in \overline{A} \Leftrightarrow$ все ветки $\overline{M}$ принимающие, а это не то, что мы хотим. Таким образом, мы приходим к следующему определению:

Определение 2.5. $\mathbf{coNP} = \{A \in \{0,1\}^* : \overline{A} \in \mathbf{NP}\}$.

Замечание. Аналогично можно доказать, что $A \in \mathbf{coNP}$ тогда и только тогда, когда ответ вычисляется как конъюнкция всех результатов работы машины или тогда и только тогда, когда $\exists V : x \in A \Leftrightarrow \forall s V(x,s) = 1$ и V вычисляется полиномиально от длины x.

Пример. 1. $\mathsf{SAT} = \{\varphi : \exists x : \varphi(x) = 1\} \in \mathbf{NP}$.

Также имеет смысл рассмотреть двойственную задачу (к задаче опровержимости формулы):

2. $\mathsf{TAUT} = \{\varphi : \exists x : \varphi(x) = 1\} \in \mathbf{coNP}$

Замечание. Несмотря на доказанную теорему о полноте, вывод не будет являться сертификатом. Действительно, вывод, вообще говоря, не обязан быть полиномиальным и, в таком случае, машина не сможет полностью его прочесть (т.к. работает полиномиально от $|x|$)

Отдельный интерес у людей науки представляет множество $(\mathbf{coNP} \cap \mathbf{NP}) \setminus \mathbf{P}$. Рассмотрим следующую задачу:

Определение 2.6. $\mathsf{FACTORING} = \{(n,a,b) : \exists d \in (a,b) : d$ — простое и $n \vdots d\}$

Утверждение 2.2. $\mathsf{FACTORING} \in \mathbf{NP} \cap \mathbf{coNP}$.

Доказательство.

$\in \mathbf{NP}$ d — сертификат

$\in \mathbf{coNP}$ сертификат — разложение на простые, каждое из которых $\notin (a, b)$.

□

2.2 Некоторые следствия из определений

Утверждение 2.3. $\mathbf{P} = \mathbf{NP} \Leftrightarrow \mathbf{P} = \mathbf{coNP}$

Утверждение 2.4. *Следует из того, что* $\mathbf{coP} = \mathbf{P}$ *(*$\mathbf{P}$ *замкнут относительно дополнения).*

Замечание. Тем не менее, может быть, что $\mathbf{P} \neq \mathbf{NP}$, но $\mathbf{NP} = \mathbf{coNP}$

Определение 2.7. $A \leqslant_p B$ сводится по Карпу (сводится полиномиально), если $\exists$ всюду полиномиально вычислимая от $|x|$ функция $f(x)$, такая, что $x \in A \Leftrightarrow f(x) \in B$.

Определение 2.8. $\mathsf{INDSET} = \{(G, k) :$ в графе G есть антиклика из k вершин$\}$

Пример. $\mathsf{CLIQUE} \leqslant_p \mathsf{INDSET}$. Действительно, $f(G, k) = f(\overline{G}, k)$ (дополнение по ребрам).

Определение 2.9. $\mathsf{4COL} = \{G : \exists$правильная раскраска в 4 цвета$\}$

Пример. $\mathsf{4COL} \leqslant_p \mathsf{SAT}$. Мы так уже делали на матлоге, когда сводили некоторые задачи к задачам выполнимости формулы. Для каждой вершины заведем две переменные p_i, q_i, отвечающие за цвет. Нам нужно для каждого ребра записать, что две вершины, являющиеся его концами, имеют разный цвет, и взять конъюнкцию, т.е:

$$\bigwedge_{(i,j)\in E} (p_i \neq p_j) \vee (q_i \neq q_j)$$

Размер данной формулы будет полиномиальным относительно размера графа.

Замечание (Свойства $\leqslant_p$)**.**

1. $A \leqslant_p B, B \leqslant_p C \Rightarrow A \leqslant_p C$
2. $A \leqslant_p B, B \in \mathbf{P} \Rightarrow A \in \mathbf{P}$
3. $A \leqslant_p B, B \in \mathbf{NP} \Rightarrow A \in \mathbf{NP}$
4. $A \leqslant_p B \Rightarrow \overline{A} \leqslant_p \overline{B}$

Определение 2.10. Задача $B \in \mathbf{NPH}$ ($\mathbf{NP}$-трудной), если $\forall A \in \mathbf{NP} : A \leqslant_p B$.

Определение 2.11. $\mathbf{NPC} = \mathbf{NP} \cap \mathbf{NPH}$ ($\mathbf{NP}$-полные)

Следствие.

1. $B \in \mathbf{NPH}, B \leqslant_p C \Rightarrow C \in \mathbf{NPH}$
2. $B \in \mathbf{NPC}, B \leqslant_p C, C \in \mathbf{NP} \Rightarrow C \in \mathbf{NPC}$

Утверждение 2.5.

1. $\mathbf{P} \cap \mathbf{NPH} \neq \varnothing \Rightarrow \mathbf{P} = \mathbf{NP}$
2. $\mathbf{coNP} \cap \mathbf{NPH} \neq \varnothing \Rightarrow \mathbf{NP} = \mathbf{coNP}$

Доказательство.

2. $B \in \mathbf{NPH}, \mathbf{coNP} \Rightarrow \overline{B} \in \mathbf{NP}$. Теперь, если $A \leqslant_p B \Rightarrow \overline{A} \leqslant_p \overline{B}$. Отсюда получаем, что $\overline{A} \in \mathbf{NP}$ и тогда $\mathbf{NP} \subset \mathbf{coNP}$. Тогда:

$$S \in \mathbf{coNP} \Rightarrow \overline{S} \in \mathbf{NP} \Rightarrow \overline{S} \in \mathbf{coNP} \Rightarrow S \in \mathbf{NP}$$

□

Утверждение 2.6. $A \in \mathbf{P}, B, \overline{B} \neq \varnothing \Rightarrow A \leqslant_p B$

Доказательство. Рассмотрим

$$f(x) = \begin{cases} \in B, x \in A \\ \notin B, x \notin A \end{cases}$$

□

Следствие. $\mathbf{P} = \mathbf{NP} \Rightarrow \mathbf{NPC} = P \setminus \{\varnothing, \Sigma^*\}$

Определение 2.12. $\mathsf{TMSAT} = \{(M, x, 1^t) : \exists y \;\; M(x, y)$ и работает за $\leqslant t$ шагов$\}$.

Утверждение 2.7. $\mathsf{TMSAT} \in \mathbf{NP}$

Доказательство. Сертификат — y, верификатор — УМТ □

Утверждение 2.8. $\mathsf{TMSAT} \in \mathbf{NPC}$

Доказательство. Принадлежность **NP** уже доказали, докажем принадлежность **NPC**. Пусть $A \in \mathbf{NP}$. По определению: $x \in A \Leftrightarrow \exists s : V(x, s) = 1$. Положим M — машину Тьюринга, вычисляющую V, $t(n)$ — время работы V для $|x| = n$. Тогда положим $f(x) = (M, x, 1^{t(|x|)})$ и получим, что $A \leqslant_p \mathsf{TMSAT}$. □