Полиномиальная сводимость. Класс $\mathcal{NP}$-complete. Семинар 4. 28 февраля 2019 г.

Подготовил: Горбунов Э.

Ключевые слова: полиномиальная сводимость, классы $\mathcal{NP}$-hard и $\mathcal{NP}$-complete, теорема Кука-Левина

Литература: [Кормен 1, Глава 36], [Кормен 3, Глава 34], [ДПВ, Глава 8]

Вспоминаем предыдущий семинар: полиномиальная сводимость, $\mathcal{NP}$-complete

Оказывается, что в классе $\mathcal{NP}$ есть «наиболее сложные» задачи, т. е. такие, к которым *сводятся* все задачи из $\mathcal{NP}$. Например, это означает, что если хоть одну из таких задач можно решать за полиномиальное время, то и любую задачу из $\mathcal{NP}$ можно решать за полиномиально время, т .е. что $\mathcal{P} = \mathcal{NP}$. Определим теперь, что мы будем понимать под сводимостью.

Определение. Говорят, что *язык A сводится полиномиально по Карпу к языку B* (и пишут $A \leq_p B$), если существует такая полиномиально вычислимая функция f, что

$$\forall x \in \Sigma^* \ (x \in A \iff f(x) \in B)\,.$$

Определение. Говорят, что *язык A сводится полиномиально по Куку к языку B* (и пишут $A \leq_T B$), если существует МТ с полиномиальной временной сложностью с оракулом для языка B, которая разрешает язык A (оракул работает за 1 такт).

Далее мы будем в подавляющем большинстве случаев работать со сводимостью по Карпу.

Определение. Говорят, что $L \in \mathcal{NP}$-hard, если $\forall A \in \mathcal{NP} \hookrightarrow A \leq_p L$.

Определение. Говорят, что $L \in \mathcal{NP}$-complete, если $L \in \mathcal{NP}\text{-hard} \cap \mathcal{NP}$.

Следующая теорема показывает, что существуют $\mathcal{NP}$-полные задачи.

Теорема 1. (Теорема Кука-Левина). Язык SAT$\in \mathcal{NP}$-complete.

Пусть $L_1 \leq_P L_2$. Тогда справедливы следующие импликации.

(i) $L_2 \in \mathcal{P} \Longrightarrow L_1 \in \mathcal{P}$;

(ii) $L_1 \notin \mathcal{P} \Longrightarrow L_2 \notin \mathcal{P}$;

(iii) $L_2 \in \mathcal{NP} \Longrightarrow L_1 \in \mathcal{NP}$.

Рассмотрим простейший пример, иллюстрирующий, как работает сводимость по Карпу.

Пример. Пусть L_1 — язык описаний графов, у которых количество рёбер не меньше числа вершин, а язык L_2 — язык описаний графов, у которых количество рёбер не меньше половины от числа вершин. Покажем, что $L_1 \leq_P L_2$. Пусть задан граф $G = (V, E)$, где V — множество вершин, E — множество рёбер. Сводящая функция будет из графа G делать граф $\tilde{G}$, состоящий из двух частей: графа G и дополнительных $|V|$ изолированных вершин. Тогда рёбер в графе $\tilde{G}$ столько же, сколько и в G, а вот вершин у $\tilde{G}$ в два раза больше. Тогда, если у исходного графа рёбер было не меньше числа вершин, то у полученного графа рёбер будет не меньше половины от числа вершин нового графа (то есть $G \in L_1 \Longrightarrow \tilde{G} \in L_2$). В обратную сторону: если у графа $\tilde{G}$ рёбер не меньше половины от числа вершин $\tilde{G}$, то у него рёбер $\geq |V|$. Но у графа G рёбер столько же, а значит, $|E| \geq |V|$ (иными словами, $G \in L_1 \Longleftarrow \tilde{G} \in L_2$), что и требовалось доказать.

Классические примеры $\mathcal{NP}$-полных языков

1. **ВЫПОЛНИМОСТЬ (SAT).** Самый известный NP-полный язык — это, конечно, ВЫПОЛНИМОСТЬ, который состоит из кодировок всех выполнимых булевых формул. Иначе говоря, для каждой формулы из языка ВЫПОЛНИМОСТЬ существуют такие значения переменных, при которых эта формула истинна. Можно считать формулы не произвольными, а, например, КНФ или даже 3-КНФ, у которых в каждый дизъюнкт входит не более 3 переменных. В последнем случае получаем язык 3-ВЫПОЛНИМОСТЬ. Можно дополнительно предполагать, как это делается в **[Кормен 1]** или **[Кормен 2]**, что *в каждый дизъюнкт входит ровно три литерала* и что *все литералы в каждом дизъюнкте 3-КНФ различны*. Но от этого требования можно и отказаться, если окажется проще строить какие-то сводимости, т. е. рассмотреть более широкий полный язык, в котором литералы в дизъюнктах могут повторяться и в каждый дизъюнкт входит не более трех литералов. **Такой трактовки языка 3-ВЫПОЛНИМОСТЬ мы и будем придерживаться в этом задании**. Тогда при часто используемом преобразовании 3-КНФ в РОВНО-3-КНФ можно просто дополнить дизъюнкт нужным числом литералов. Например, дизъюнкт $\neg x_2 \vee x_3$ переписывается в эквивалентном виде $\neg x_2 \vee x_3 \vee x_3$ или $\neg x_2 \vee x_3 \vee \neg x_2$. Другое дело, что некоторые сводимости при таком понимании 3-КНФ, возможно, перестанут выполняться, и тогда нужно уточнить и/или изменить сами сводимости.

2. **ПРОТЫКАЮЩЕЕ МНОЖЕСТВО.** Дано семейство конечных множеств $\{A_1, \ldots, A_m\}$ и натуральное число k. Существует множество мощности k, пересекающее *каждое* A_i. Язык остается NP-полным, *даже если предположиить, что мощности всех* A_i *равны* 2.

3. **КЛИКА.** Даны неориентированый граф G и натуральное число k. В G есть клика (полный подграф) на k вершинах.

4. **ВЕРШИННОЕ ПОКРЫТИЕ.** Даны неориентированый граф $G = (V, E)$ и натуральное число k. В G есть *вершинное покрытие* мощности k, т. е. такое подмножество вершин $V' \subseteq V$ мощности k, что хотя бы *один конец каждого ребра* входит в V'.

5. **НЕЗАВИСИМОЕ МНОЖЕСТВО.** Даны неориентированый граф $G = (V, E)$ и натуральное число k. В G есть *независимое множество* мощности k, т. е. такое подмножество вершин $V' \subseteq V$ мощности k, что никакие два из них не соединены ребром.

6. **ХРОМАТИЧЕСКОЕ ЧИСЛО.** Даны неориентированый граф G и натуральное число $k > 2$. Вершины G можно раскрасить в k цветов так, чтобы смежные вершины были окрашены в разные цвета. При $k = 3$ получаем язык **3-COLOR**, и он также NP-полный. Отметим, что язык 2-COLOR уже можно полиномиально разрешить (казалось бы, совсем немного изменили условие).

7. **ГАМИЛЬТОНОВ ЦИКЛ (ГЦ)**. Дан неориентированый граф G, в котором есть *гамильтонов цикл*. Иными словами, существует циклический обход всех вершин графа, не попадающий ни в какую вершину дважды.

8. **ГАМИЛЬТОНОВ ПУТЬ (ГП)**. Дан неориентированый граф G, в котором есть *гамильтонов путь*. Иными словами, существует путь, который проходит через каждую вершину графа ровно один раз.

9. **РАЗБИЕНИЕ** или **ЗАДАЧА О КАМНЯХ**. Дано конечное множество (куча) камней A, причем вес каждого камня $a \in A$ является целым положительным числом $s(a)$. Можно разбить A на две кучи одинакового веса. Иными словами, существует такое подмножество $A' \in A$, что $\sum_{a \in A'} s(a) = \sum_{a \in A \setminus A'} s(a)$.

10. **3-СОЧЕТАНИЕ**. Дано множество $M \subseteq W \times X \times Y$, где W, X и Y — непересекающиеся множества, содержащие одинаковое число элементов q. В M есть *трехмерное сочетание*, т. е. такое подмножество $M' \subseteq M$ мощности q, никакие два элемента которого не имеют ни одной одинаковой координаты.

11. **РЮКЗАК**. Дана натуральные числа $\{a_1, \ldots, a_n\}$ и натуральное число b, такие что сумма некоторых a_i равна b.

12. **max − 2-ВЫПОЛНИМОСТЬ**. Дана 2-КНФ (т. е. КНФ, в каждую дизъюнкцию которой входит не более двух логических переменных) и двоичное число k. Существует такой набор значений логических переменных, что выполняются k или более дизъюнкций.

13. **МАКСИМАЛЬНЫЙ РАЗРЕЗ**. Дан граф G и натуральное число k. Множество вершин графа можно разбить на два непересекающихся подмножества, между которыми можно провести не менее k ребер.

    Иногда говорят о взвешенном варианте задачи. Дан граф $G(V, E)$ с неотрицательной весовой функцией на ребрах $w : E \to \mathbb{Z}_+$ и натуральное число k. Можно найти дизъюнктное разбиение множества $V = V_1 \sqcup V_2$, такое что сумма весов ребер, соединяющих V_1 и V_2, не менее k.

14. **N[ot]A[ll]E[qual]-SAT**. Дана КНФ-формула, для которой существует набор, такой что в каждом дизъюнкте есть истинный и ложный литералы.

15. **ТРЁХДОЛЬНОЕ СОЧЕТАНИЕ.** Язык трёхдольных графов (каждая из долей имеет размер $3n$, вершины одной доли ребром не соединены) на $3n$ вершинах, в котором есть n, непересекающихся треугольников.

16. **УРАВНЕНИЯ В НУЛЯХ И ЕДИНИЦАХ (УНЕ).** Язык битовых матриц $\mathbf{A}$ (из нулей и единицы) таких, что существует такой вектор $\mathbf{x}$ из нулей и единиц, что $\mathbf{Ax} = \mathbf{1}$, где $\mathbf{1}$ — вектор из единиц.

Теперь построим некоторые сводимости между указанными языками.

Пример. ГП $\leq_P$ ГЦ. Сводящая функция по графу G строит граф $\tilde{G}$, который является копией графа G + в графе $\tilde{G}$ добавляется новая вершина (обозначим её u), которая соединена со всеми остальными вершинами. Если в исходном графе G существовал гамильтонов путь, то в построенном графе $\tilde{G}$ мы можем рассмотреть этот же путь и достроить его до цикла, если учтём, что обе концевые точки пути соединены с новой вершиной. Обратно, если $\tilde{G}$ содержит гамильтонов цикл, то найдутся две вершины (v_1, v_2) в $\tilde{G}$, которые смежны u в гамильтоновом цикле. Значит, путь, получаемый отбрасыванием вершины u из вершины v_1 в вершину v_2 является гамильтоновым в исходном графе.

Пример. ГЦ $\leq_P$ ГП. Сводящая функция по графу G (считаем, что множество вершин совпадает с $\{1, 2, \ldots, n\}$) строит граф $\tilde{G}$, который является копией графа G + добавляются 3 вершины: $n+1$, которая копирует вершину n (т.е. она соединена с теми же вершинами, что и n), и две висячие вершины $n+2$ и $n+3$, где $n+2$ соединена с n, а $n+3$ — с $n+1$. Если в графе G есть гамильтонов цикл, то рассмотрим в этом цикле вершину, предшествующую вершине n, обозначим её через u. Тогда по построению u соединена с $n+1$, а значит, в графе $\tilde{G}$ есть гамильтонов путь, который начинается в $n+2$, затем полностью повторяет путь по циклу из n в u, затем идёт в $n+1$, откуда попадает в $n+3$. Обратно, если в $\tilde{G}$ есть гамильтонов путь, то концевыми вершинами обязаны быть $n+2$ и $n+3$, т.к. у них ровно по одному ребру. Отбросим их. Тогда в оставшемся графе есть гамильтонов путь из n в $n+1$. Рассмотрим в нём предпоследнюю вершину, обозначим её через v. Так как она соединена ребром с $n+1$, а $n+1$ — это копия n, то в исходном графе эта вершина соединена с n, а значит, кусок гамильтонова пути из n в $n+1$ до вершины u + ребро (u, n) образуют гамильтонов цикл в исходном графе G.

Пример. 3-ВЫПОЛНИМОСТЬ $\leq_P$ НЕЗАВИСИМОЕ МНОЖЕСТВО. Для каждого дизъюнкта вида $a \vee b \vee c$ введём три вершины, помеченные символами a, b, c и соединённые рёбрами между собой (получится треугольник). Для дизъюнктов вида $d \vee e$ добавим две вершины, помеченные символами d и e и соединённые ребром. Для дизюнктов, состоящих из одного литерала, добавим вершину, помеченную этим литералом. Кроме того, для всех пар вершин, помеченных противоположными литералами, проведём между ними ребро. Время построения такого графа полиномиально по входу. Если исходная формула была выполнима, то в каждом наборе дизъюнктов можно выбрать по литералу, который можно приравнять единице. Это соответствует выбору независимого множества в построенном графе, причём мощность этого независимого множества равна числу дизъюнктов в булевой формуле. Так как литерал и его отрицание одновременно быть равными единице не могут, то не возникнет двух вершин из нашего множества, которые соединены ребром. В обратную сторону: пусть мы построили независимое множество в таком графе, состоящее из m вершин, где m — число дизъюнктов в исходной формуле. Тогда в каждом наборе вершин, отвечающих одному и тому же дизюнкту, мы могли взять только одну вершину, причём в нашем наборе нет вершин, отвечающих противоположным литералам. Это означает, что мы можем правильным образом присвоить значения переменных, чтобы литералы, соответствующие вершинам в нашем независимом множестве, равнялись бы единице. Иными словами мы получаем, что булева формула выполнима.

Пример. КЛИКА $\leq_P$ НЕЗАВИСИМОЕ МНОЖЕСТВО. Одна задача сводится к другой одинаково просто. Пусть дан граф $G = (V, E)$. Рассмотрим его дполонени: $\bar{G} = (V, \bar{E})$, где $\bar{E}$ — это отсутствующие в E

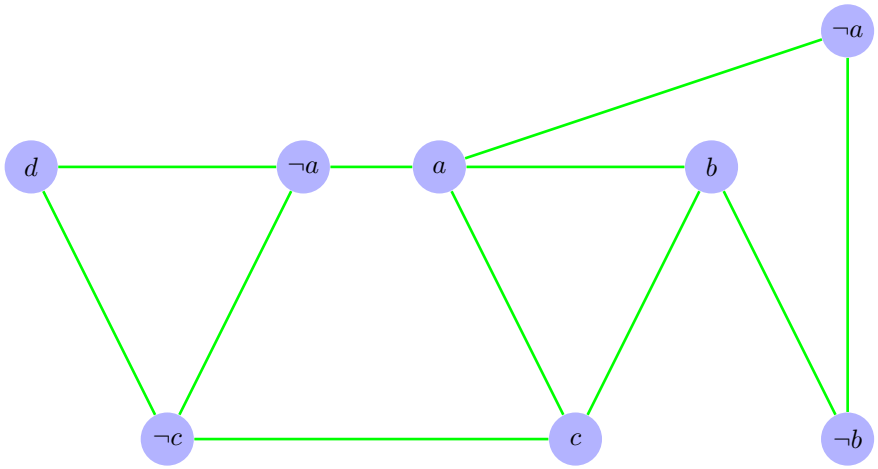

Рис. 1: Граф, построенный по формуле $(d \vee \neg a \vee \neg c) \wedge (a \vee b \vee c) \wedge (\neg a \vee \neg b)$

рёбра. Тогда в графе G существует клика на k вершинах тогда и только тогда, когда существует независимое множество мощности k в графе $\bar{G}$.

Пример. ТРЁХДОЛЬНОЕ СОЧЕТАНИЕ $\leq_P$ УНЕ. Одна из интерпретаций трёхдольного сочетания: имеется n мальчиков, n девочек и n домашних животных. Каждому мальчику нравятся какие-то девочки и какой-то набор животных, а девочкам — какой-то набор мальчиков и какой-то набор животных. Нужно проверить, если ли n троек в каждой из которых один мальчик, одна девочка и одно животное, таких что все друг другу нравятся (считаем, что если мальчику нравится девочка, то он ей тоже нравится; кроме того, считаем домашним животным тоже нравятся только те люди, которым они нравятся). Занумеруем все возможные тройки (мальчик, девочка, питомец). Пусть их m штук. Введём булевы переменные $x_1, x_2, \ldots, x_m$, где $x_i = 1$ означает, что i-я тройка вошла в сочетание. Запишем уравнение, гарантирующее, что решение, задаваемое переменными $x_1, \ldots, x_m$ является корректным сочетанием. Рассмотрим произвольного мальчика. Пусть он входит в тройки с номерами $i_1, \ldots, i_k$. Условие на то, что мальчик входит ровно в одну тройку: $x_{i_1} + x_{i_2} + \ldots + x_{i_k} = 1$. Аналогичные уравнения нужно составить для всех остальных мальчиков, девочек и животных. В итоге получим матрицу из нулей и единиц, у которой $3n$ строк и m столбцов. Найти все возможные тройки, соединённые рёбрами, самым прямолинейным способом можно за $O(n^3)$, просто перебрав все тройки вершин (мы даже сейчас не думаем, что они могли быть из одной доли) и проверив для каждой, что вершины соединены рёбрами.

Пример. НЕЗАВИСИМОЕ МНОЖЕСТВО $\leq_P$ ВЕРШИННОЕ ПОКРЫТИЕ. Заметим, что множество вершин S является вершинным покрытием графа $G = (V, E)$ тогда и только тогда, когда множество $V \setminus S$ является независимым в G. Действительно, если S — вершинное покрытие, то, если есть ребро между вершинами в $V \setminus S$, то оно не покрыто ни одной вершиной из вершинного покрытия, противоречие; если же рёбер между вершинами $V \setminus S$ нет, то все рёбра смежны с вершинами из S (иными словами, если $V \setminus S$ — независимое множество, то S — вершинное покрытие). Поэтому, чтобы решить задачу о независимом множестве на входе (G, k) достаточно решить задачу о вершинном покрытии на входе $(G, |V| - k)$.

Другие примеры сводимостей вас ждут в домашнем задании.