Дисциплина

«РАЗРАБОТКА ЭФФЕКТИВНЫХ АЛГОРИТМОВ»

Лекция 11

ОСНОВНЫЕ КЛАССЫ ТЕОРИИ СЛОЖНОСТИ ВЫЧИСЛЕНИЙ

ПОЛИНОМИАЛЬНАЯ СВОДИМОСТЬ, КЛАССЫ *NPC* И *NPH*

Лектор: Михаил Васильевич Ульянов,

muljanov@mail.ru, **8 916 589 94 04**

1. Выпуклый охватывающий контур и сортировка

Рассмотрим задачу о выпуклом охватывающем контуре:

Дано n точек с положительными координатами (x_i, y_i), $i = 1, n$. Необходимо найти выпуклый охватывающий контур на этих точках и перечислить точки контура обходом от левой нижней точки против часовой стрелки.

Я умею забивать гвоздики в фанерку по координатам (x_i, y_i), после чего умею натягивать веревку на эти гвоздики и, следовательно, получаю охватывающий контур. Но он должен быть контуром — т.е. иметь не равную нулю площадь. Затем я перечисляю точки контура от левой нижней точки против часовой стрелки.

Научите меня сортировать положительные числа — массив $\{a_i, i = 1, n\}$ через построение выпуклого охватывающего контура на множестве точек.

Для этого необходимо придумать соответствие — $a_i \rightarrow (x_i, y_i)$?

Вариант $a_i \rightarrow (a_i, {a_i}^2)$.

Таким образом, мы свели задачу сортировки к задаче о выпуклом контуре.

Вопрос: Сведение в обратную сторону возможно? Т.е., из того, что мы умеем решать задачу сортировки, следует ли, что мы умеем решаем задачу о выпуклом контуре?

Ответ: **нет**! Рассмотрите множество точек, которые не все лежат на контуре, среди этих точек есть точки, лежащие внутри контура.

2. Полиномиальная сводимость

Формально:

Мы имеем задачу Z_1 и решающий эту задачу алгоритм, выдающий правильный ответ для всех допустимых входов, и, предположим, что для задачи Z_2 алгоритм решения нам неизвестен.

Если мы можем переформулировать допустимые входы задачи Z_2 в терминах входов задачи Z_1, то мы решаем задачу Z_1 , используя известный нам алгоритм и возвращаем решение в задачу Z_2.

Пусть задача Z_1 задана множеством допустимых входов D_{Z1}, а задача Z_2 — множеством D_{Z2}, и существует функция f_s, реализованная некоторым алгоритмом, сводящая каждую конкретную проблему (вход) $d_{Z2} \in D_{Z2}$ задачи Z_2 к конкретной проблеме (входу) $d_{Z1} \in D_{Z1}$ задачи Z_1, т. е. если

$$\forall d_{Z2} \in D_{Z2} \;\; f_s(d_{Z2} \in D_{Z2}) = d_{Z1} \in D_{Z1},$$

то задача Z_2 сводима к задаче Z_1.

Если при этом временная сложность алгоритма, реализующего функцию f_s есть $O(n^k)$, где n — длина входа, а k — некоторая константа, т. е. для задачи сведения предъявлен полиномиальный сертификат, т.е. задача сведения принадлежит классу P, то тогда говорят, что задача Z_2 полиномиально сводится (сводима) к задаче Z_1.

Таким образом, полиномиальная сводимость равнозначна принадлежности задачи сведения к классу P.

Обозначение:

$$Z_2 \underset{P}{\rightarrow} Z_1$$

3. Теорема Stephen Cook, 1971

Формулировка:

$$\exists Z^* \in NP: \ \ \forall Z \in NP \ \ Z \underset{P}{\to} Z^*$$

Это означает, что существует такая задача (и эта теорема конструктивна, т.е. она ее явно указывает), к которой полиномиально сводятся все задачи класса NP, абсолютно все задачи (!), как известные нам, так и не известные. Теорема требует только полиномиальной проверяемости решения задачи, т.е. ее принадлежности к классу NP!

4. Класс NPC или класс NP -полных задач.

Понятие NP-полноты было введено независимо Куком (Stephen Cook, 1971) и Левиным (1973). Определение класса NPC ($NPcomplete$) или класса NP-полных задач требует выполнения следующих двух условий:

— во-первых, задача должна принадлежать классу NP,

— во-вторых, к этой задаче должны полиномиально сводиться *все* задачи из класса NP, что схематично представлено на рисунке 1.

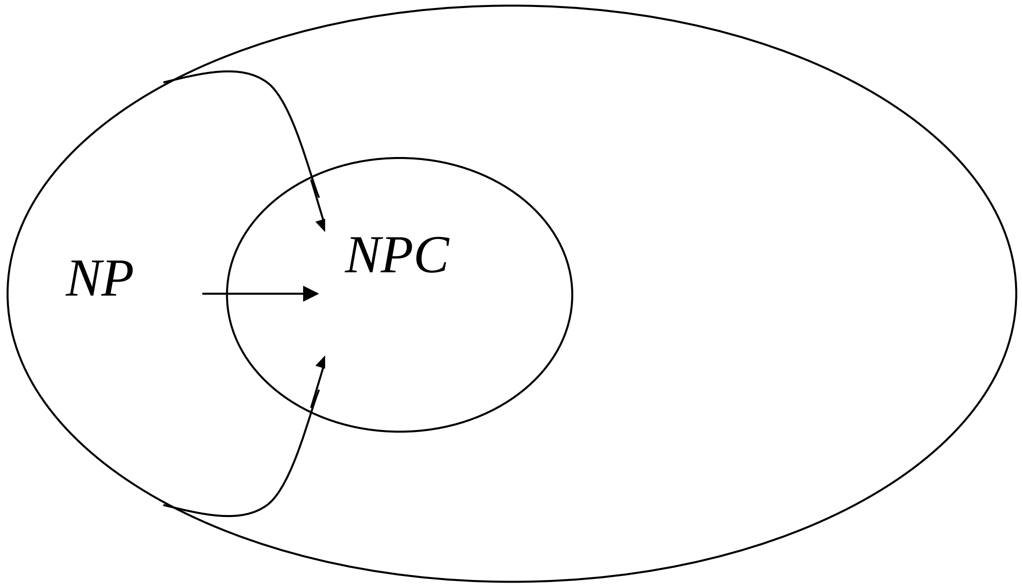

Рис. 1. Сводимость задач и класс NPC

Для класса NPC доказана следующая теорема: если существует задача, принадлежащая классу NPC, для которой существует полиномиальный алгоритм решения, то класс P совпадает с классом NP, т. е. $P = NP$. Схема доказательства состоит в сведении с полиномиальной трудоемкостью любой задачи из класса NP к данной задаче из класса NPC и решении этой задачи за полиномиальное время (по условию теоремы). Сумма двух полиномов есть полином!

В настоящее время доказано существование сотен NP-полных задач, но ни для одной из них пока не удалось найти полиномиального алгоритма решения. Отметим, что для многих из них предложены приближенные полиномиальные алгоритмы. Сегодня ученые предполагают, что соотношение классов, имеет вид, показанный на рис. 2, а именно $P \neq NP$, то есть $NP \backslash P \neq 0$, и ни одна задача из класса NPC не может быть решена, по крайней мере, сегодня, с полиномиальной сложностью.

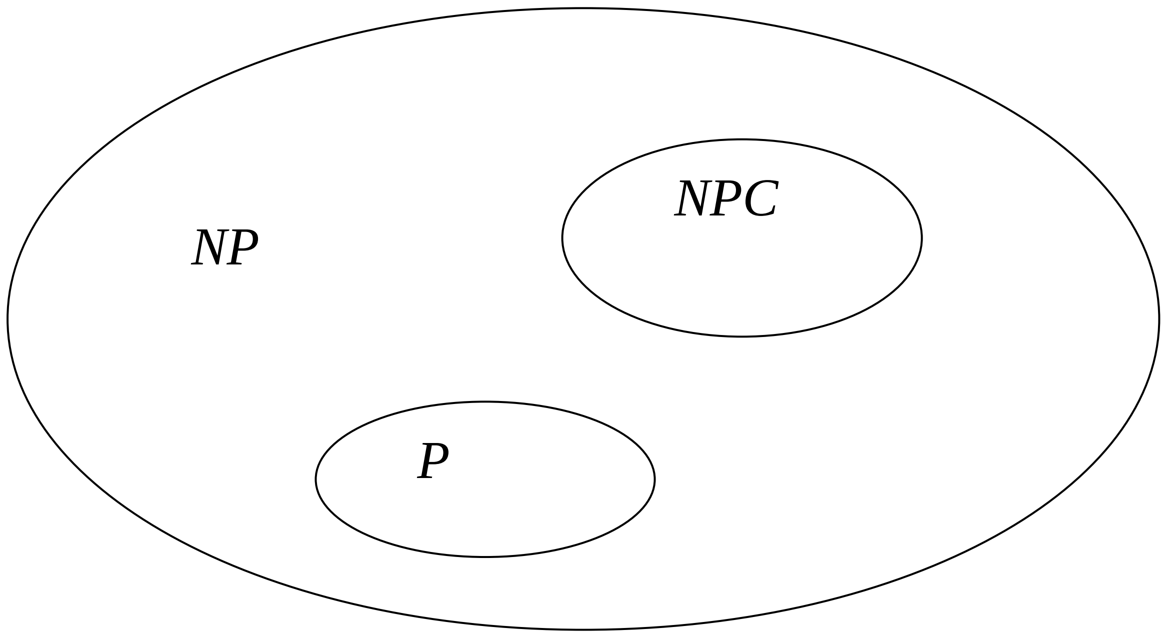

Рис. 2. Соотношение классов P, NP, NPC

Исторически первое доказательство NP-полноты было предложено Куком для задачи о выполнимости схемы (теорема Кука). Метод сведения за полиномиальное время был предложен Карпом (Richard Karp) и использован им для доказательства NP-полноты целого ряда задач. Описание многих NP-полных задач содержится в классической книге Гэри и Джонсона, отметим, что задача о сумме также является NP-полной задачей.

5. Задача Z^* (теорема Кука)

Есть схема из функциональных элементов (И, ИЛИ, НЕ), у которой n входов и один выход. Число функциональных элементов ограничено как $O(n^k)$. Есть ли такой вход схемы, для которого выход — «1»? На вход алгоритма подается описание схемы на каком-то выбранном языке (VHDL). Решение — вектор входа схемы, при котором на выходе будет — «1».

Почему эта задача принадлежит классу NP?

Решение прямым перебором — всего есть 2^n вариантов входных векторов, для каждого из которых необходимо выполнить схему за $O(n^k)$ операций.

6. Класс NPH — NP-трудные задачи

Мы знаем, что классом NP не ограничивается универсум задач. Вне класса NP находятся задачи, решение которых не может быть полиномиально проверено. Сюда относится большая группа задач дискретной оптимизации — задачи о расписаниях, задача коммивояжера и т.д. Отметим, что задача линейного программирования относится к классу P (метод вписанных эллипсоидов – Хачиян).

Построение класса NPH рассмотрим на примере задачи коммивояжера. Принадлежность задачи к классу NPH доказывается следующими двумя этапами:

1. Исходная задача (коммивояжер) переформулируется как задача разрешения. Т.е. вместо задачи поиска минимального по стоимости тура рассматривается задача нахождения тура со стоимостью меньше наперед заданной. Этот тур не обязан быть оптимальным! Очевидно, что задача разрешения для задачи коммивояжера принадлежит классу NP — полученное решение полиномиально проверяемо. Тур состоит из n дуг, и проверка включает суммирование их весов — $\Theta(n^1)$.

2. Путем сведения какой-либо задачи из NPC к полученной задаче разрешения доказывается, что задача разрешения для задачи коммивояжера лежит в NPC.

Эти два доказательства и приводят к тому, что исходная задача — задача коммивояжера лежит в классе NPH, т.е. является NP-трудной задачей.

Список литературы к лекции 11

[1.] Хопкрофт Дж., Мотовани Р., Ульман Дж. Ведение в теорию автоматов, языков и вычислений, 2-е изд.: Пер. с англ. — М.: Издательский дом «Вильямс», 2002. — 528 с.

[2.] Носов В. А. Основы теории алгоритмов и анализа их сложности (http: // intsys.msk.ru).

[3.] Бондаренко В. А. О сложности дискретных задач (http: //edu.yar.ru/russian /pedbank/sor-pro/bondarenko).

[4.] Макконелл Дж. Основы современных алгоритмов. 2-е дополненное издание. — М.: Техносфера, 2004. — 368 с.

[5.] Cobham A. The intrinsic computational difficulty of functions // Proc. Congress for Logic, Mathematics, and the Philosophy of Science. – North Holland, Amsterdam, 1964. pp. 24–30.

[6.] Cook S. C. The complexity of theorem–proving procedures // Third ACM Symposium on Theory of Computing., — ACM, New York, 1971. pp. 151–158.

[7.] Karp R. M. Reducibility among combinatorial problems // Complexity of Computer Computations / R. E. Miller, ed., – Plenum Press, New York, 1972. pp. 85–104.

[8.] Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ, 2-ое издание: Пер. с англ. — М.: Издательский дом «Вильямс», 2005. — 1296 с.

[9.] Левин Л. А. Универсальные проблемы упорядочения // Проблемы передачи информации. 1973. — Т. 9. № 3. С. 115–117.

[10.] Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. — М.: Мир, 1982. — 416 с.