Вычислительная сложность: основные понятия, типовые результаты и открытые вопросы

А. А. Агеев

Институт математики им. С. Л. Соболева
Новосибирск

Верификационная задача

Decision Problem

Верификационная задача (Decision Problem) состоит из входа и вопроса. Решение - ответ "да"или "нет".

Subset Sum Problem

Вход: конечное множество целых чисел T.
Вопрос: содержит ли T подмножество чисел, сумма которых равна нулю?

Пример: для множества $T = \{-7, -3, -2, 5, 8\}$ ответ ”да“, ибо подмножество $\{-3, -2, 5\}$ в сумме даёт ноль.
Для множества $T = \{-4, -3, 8, 10\}$ ответ “нет”.

Класс NP

Вход верификационной задачи - слово.
Множество примеров верификационной задачи, для которых ответ "да", можно рассматривать как язык над конечным алфавитом.
Класс NP определяется для множество языков над конечным алфавитом Σ.

Определение класса NP

Язык L называется *принадлежащим классу* NP, если существует двухместный предикат $R(x, y)$, вычислимый за полиномиальное время, и константа $c > 0$ такие, что для всякого слова x условие "x принадлежит L" равносильно условию "найдется слово y длины меньше $|x|^c$ такое, что верно $R(x, y)$".
Слово y называется *сертификатом принадлежности* слова x к языку L.

Класс NP

Subset Sum Problem

Вход: конечное множество целых чисел T.
Вопрос: содержит ли T подмножество чисел, сумма которых равна нулю?

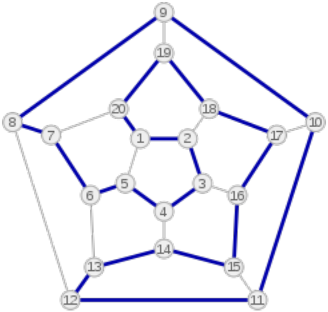
Определение класса NP

Язык L называется *принадлежащим классу* NP, если существует двухместный предикат $R(x, y)$, вычислимый за полиномиальное время, и константа $c > 0$ такие, что для всякого слова x условие "x принадлежит L" равносильно условию "найдется слово y длины меньше $|x|^c$ такое, что верно $R(x, y)$".
Слово y называется *сертификатом принадлежности* слова x к языку L.

Hamiltonian Cycle

Вход: неориентированный граф G.
Вопрос: содержит ли G гамильтонов цикл?

Пример 1:



Hamiltonian Cycle

Вход: неориентированный граф G.
Вопрос: содержит ли G гамильтонов цикл?

Пример 2 (задача о нахождении маршрута шахматного коня, проходящего через все поля шахматной доски по одному разу):

38	35	62	25	60	23	10	7
63	26	37	34	11	8	59	22
36	39	28	61	24	57	6	9
27	64	33	40	5	12	21	58
50	29	4	13	48	41	56	19
1	14	49	32	53	20	47	44
30	51	16	3	42	45	18	55
15	2	31	52	17	54	43	46

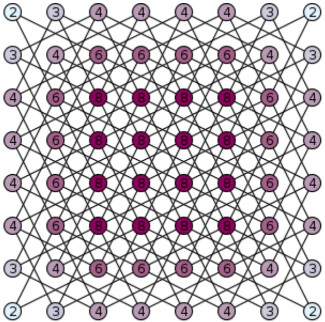
A Closed Knight's Tour of the 8×8 Chessboard

Задачи из класса NP

Hamiltonian Cycle

Вход: неориентированный граф G.
Вопрос: содержит ли G гамильтонов цикл?

Пример 2 (граф, вершинами которого являются поля доски, и два поля соединены ребром, если с одного можно попасть на другое за один ход коня):



Отступление

Задачу о нахождении обхода конём для обычной доски из математиков первым исследовал Эйлер в работе «Решение одного любопытного вопроса, который, кажется, не подчиняется никакому исследованию» (1757 год).
Для произвольных прямоугольных досок задача полностью решена сравнительно недавно.

In 1991 Schwenk [1] completely answered the question: Which rectangular chessboards have a knight's tour?

Theorem 1 (Schwenk) *An $m \times n$ chessboard with $m \leq n$ has a closed knight's tour unless one or more of the following three conditions hold:*

(a) m and n are both odd
(b) $m \in \{1, 2, 4\}$;
(c) $m = 3$ and $n \in \{4, 6, 8\}$.

История

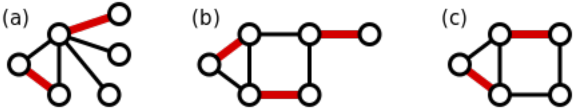
Название «гамильтонов цикл» произошло от задачи «Кругосветное путешествие» предложенной ирландским математиком Вильямом Гамильтоном в 1859 году. Нужно было, выйдя из исходной вершины графа, обойти все его вершины и вернуться в исходную точку. Граф представлял собой укладку додекаэдра, каждой из 20 вершин графа было приписано название крупного города мира.

Паросочетанием в неориентированном графе называется множество из попарно несмежных рёбер.
Паросочетание называется *совершенным*, если оно покрывает все вершины.

Perfect Matching

Вход: неориентированный граф G.
Вопрос: содержит ли G совершенное паросочетание?

Пример:



Задачи из класса NP

Perfect Matching

Вход: неориентированный граф G.
Вопрос: содержит ли G совершенное паросочетание?

Содержательная постановка: Имеется $2k$ студентов и k двухместных комнат в общежитии. Про каждую пару студентов известно, являются ли они приятелями или нет.Спрашивается, можно ли расселить всех студентов по комнатам таким образом, чтобы в каждой комнате оказались приятели.

Задачи из класса NP

Composite Number

Вход: Целое положительное число N.
Вопрос: Является ли N составным?

Задачи из класса NP

Изоморфизмом графов $G = \langle V_G, E_G \rangle$ и $H = \langle V_H, E_H \rangle$ называется биекция между множествами вершин графов $f\colon\ V_G \to V_H$ такая, что любые две вершины u и v графа G смежны тогда и только тогда, когда вершины $f(u)$ и $f(v)$ смежны в графе H.

Graph Isomorphism

Вход: Неориентированные графы G и H.
Вопрос: Изоморфен ли граф G графу H?

Конъюнктивная нормальная форма - булева формула в виде конъюнкции дизъюнкций литералов.
Пример: $(\overline{x}_1 \vee x_2 \vee x_3)\&(x_2 \vee \overline{x}_3 \vee x_4)$.

Satisfiability (SAT)

Вход: Конъюнктивная нормальная форма $F(x_1, \ldots, x_n)$.
Вопрос: Существует ли набор значений переменных $x_1, \ldots, x_n$, для которого $F(x_1, \ldots, x_n) = \mathbf{1}$.

NP-полная задача

Пусть L_1 и L_2 - два языка над алфавитом Σ. Язык L_1 *полиномиально сводится (сводится по Карпу)* к языку L_2, если существует функция $f : \Sigma^* \to \Sigma^*$, вычислимая за полиномиальное время и обладающая следующим свойством:

$x \in L_1$ тогда и только тогда, когда $f(x) \in L_2$.

Сводимость по Карпу обозначается как $L_1 \propto L_2$.

Определение NP-полной задачи

Язык (=верификационная задача) L называется NP-полным, если он лежит в классе NP и любой язык из класса NP полиномиально сводится к нему.

NP-полная задача

Теорема Кука-Левина (1971)

Задача **SAT** NP-полна.

Следствие

Если задача **SAT** полиномиально разрешима, то и любая задача из класса NP полиномиально разрешима, то есть NP=P.

NP-полные задачи

Для того чтобы доказать, что какая-то задача из класса NP NP-полна, достаточно свести к ней NP-полную задачу.

Список Карпа - список, состоящий из формулировок и доказательств NP-полноты 21-й задачи, опубликованный в

Richard M. Karp (1972). "Reducibility Among Combinatorial Problems". In R. E. Miller and J. W. Thatcher (editors). Complexity of Computer Computations. New York: Plenum. pp. 85–103.

Классическая монография по вычислительной сложности:

Garey, Michael R.; Johnson, David S. (1979), Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman

NP-полные задачи

Для того чтобы доказать, что какая-то задача из класса NP NP-полна, достаточно свести к ней NP-полную задачу.

Subset Sum NP-полна (Karp 1972).

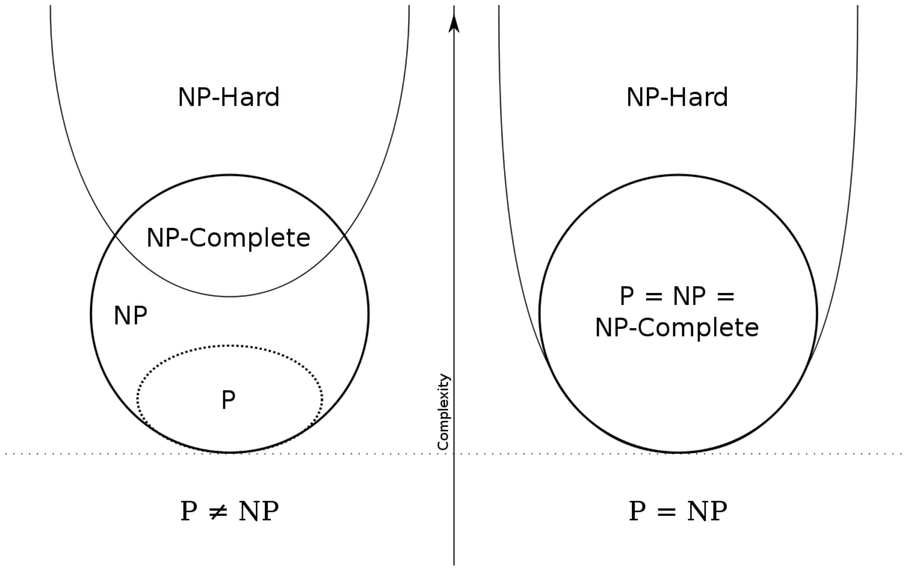
Hamiltonian Cycle NP-полна (Karp 1972).

Perfect Matching полиномиально разрешима (Edmonds 1965).

Composite Number полиномиально разрешима (Agrawal, Kayal & Saxena 2004).

Graph Isomorphism вопрос открытый.

Проблема P versus NP



Проблема P versus NP

Первая проблема в списке Millenium Prize Problems:

P versus NP problem
Hodge conjecture
Poincare conjecture (solved)
Riemann hypothesis
Yang–Mills existence and mass gap
Navier–Stokes existence and smoothness
Birch and Swinnerton-Dyer conjecture

Проблема P versus NP

Десятки "доказательств"P=NP и P≠NP можно найти на сайте Gerhard Woeginger http://www.win.tue.nl/ gwoegi/P-versus-NP.htm

Серьезная попытка Vinay Deolalikar (2010). Neil Immerman обнаружил фатальные ошибки.

Задачи дискретной оптимизации

Traveling Salesman

Вход: полный неориентированный граф G, неотрицательные длины рёбер.
Найти: гамильтонов цикл минимальной длины.

Пример: оптимальный маршрут коммивояжёра через 15 крупнейших городов Германии. Указанный маршрут является самым коротким из всех возможных 43 589 145 600.

Traveling Salesman

In March 2005, the travelling salesman problem of visiting all 33,810 points in a circuit board was solved using Concorde TSP Solver: a tour of length 66,048,945 units was found and it was proven that no shorter tour exists. The computation took approximately 15.7 CPU-years (Cook et al. 2006). In April 2006 an instance with 85,900 points was solved using Concorde TSP Solver, taking over 136 CPU-years, see Applegate et al. (2006).
In March 2005, the travelling salesman problem of visiting all 33,810 points in a circuit board was solved using Concorde TSP Solver: a tour of length 66,048,945 units was found and it was proven that no shorter tour exists. The computation took approximately 15.7 CPU-years (Cook et al. 2006). **In April 2006 an instance with 85,900 points was solved using Concorde TSP Solver, taking over 136 CPU-years, see Applegate et al. (2006).**

The Concorde TSP Solver is a program for solving the traveling salesman problem. It was written by David Applegate, Robert E. Bixby, Vacek Chvatal, and William J. Cook, in ANSI C, and is freely available for academic use.

Задачи дискретной оптимизации

Формальное определение

Задача дискретной оптимизации - четвёрка (I, Sol, w, g), где

- I - множество примеров;
- $Sol(x)$ - множество допустимых решений примера $x \in I$;
- для $x \in I$ и $y \in Sol(x)$ $w(x, y)$ - вес y.
- g - целевая функция, либо $\min$, либо $\max$.

Цель - для заданного примера $x \in I$ среди всех допустимых решений найти решение y минимального или максимального веса (размера, длины, стоимости), то есть,

$$w(x, y) = g\{w(x, y') \mid y' \in Sol(x)\}.$$

Решение y называется оптимальным.

Класс NPO

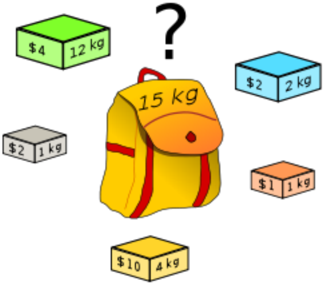
NPO - класс задач дискретной оптимизации, удовлетворяющих следующим условиям:

- размер каждого допустимого решения ограничен полиномом от размера примера;
- языки $\{x \mid x \in I\}$ и $\{(x, y) \mid y \in Sol(x)\}$ распознаются за полиномиальное время.
- w вычислима за полиномиальное время.

Для задач из класса NPO верификационные версии принадлежат классу NP.
Верификационная версия - задача с тем же входом плюс некоторое число K.
Требуется установить, существует ли допустимое решение веса не более (не менее), чем K.
Если верификационная версия NP-полна, то исходная оптимизационная задача NP-трудна.

Knapsack Problem

Пусть имеется набор предметов, каждый из которых имеет два параметра — вес и стоимость. И есть рюкзак определённой вместимости. Задача заключается в том, чтобы собрать рюкзак с максимальной стоимостью предметов внутри, соблюдая при этом весовое ограничение рюкзака.



Knapsack Problem

Имеется n предметов $\{z_1 \dots z_n\}$, каждый предмет z_i имеет ценность v_i и вес w_i.
Через x_i обозначим число копий предмета z_i, $x_i \in \{0, 1\}$. Максимальный вес, который мы можем унести, равен W.
Максимизировать

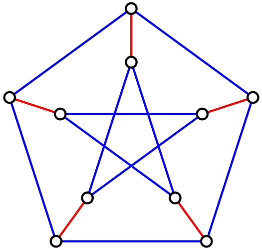
$$\sum_{i=1}^{n} v_i x_i \text{ при ограничении } \qquad \sum_{i=1}^{n} w_i x_i \leqslant W, \qquad x_i \in \{0, 1\}.$$

Knapsack Problem NP-трудна, но может быть решена динамическим программированием за время $O(nW)$.

Maximum weighted matching

Вход: полный неориентированный граф G, неотрицательные веса на рёбрах.
Найти: паросочетание максимального веса.

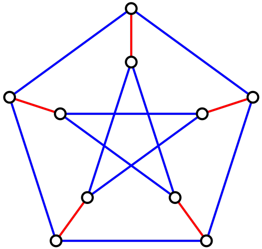
Maximum weighted matching полиномиально разрешима (Edmonds,1965).



Maximum weighted matching

Вход: полный неориентированный граф G, неотрицательные веса на рёбрах.
Найти: паросочетание максимального веса.

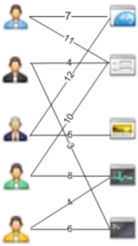
Maximum weighted matching полиномиально разрешима (Edmonds,1965).



Assignment Problem

Вход: полный двудольный граф G с равными размерами долей, неотрицательные веса на рёбрах.
Найти: совершенное паросочетание минимального веса.

Имеется равное число работ и исполнителей. Любой исполнитель может быть назначен на выполнение любой (но только одной) работы, но с неодинаковыми затратами. Нужно распределить работы по исполнителям так, чтобы каждый исполнитель получил работу, и все работы были выполнены с минимальными суммарными затратами.



La Fine

Благодарю за внимание!