Теорема Кука-Левина

SAT – NP-полна

Док-во

- SAT – задача с полином проверкой ⇒ (по теореме) SAT лежит в NP.

- Построим сведение:

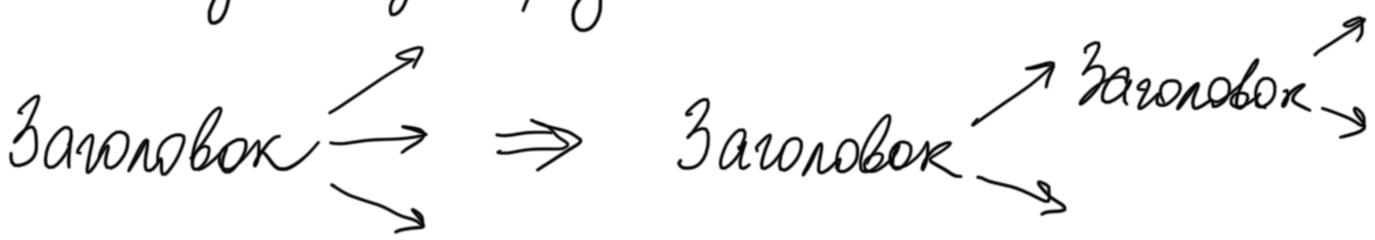
Пусть A – произвольная задача из NP, надо доказать, что $A \leq_p SAT$

Есть МТ, решающая A. В построении используем её (она не детерменирована и решает A за время $p(n)$, где n – длина экземпляра задачи A).

Будем считать, что у этой НМТ для каждого заголовка, для которого есть команда, есть ровно две команды с таким заголовком.

Пусть теперь для этого заголовка есть больше команд, тогда переорганизуем это в бинарное дерево.

Наша НМТ – бинарная, т.е. выбирает одну из двух команд каждый раз.



то есть :

$$\begin{tikzcd}
 & q_k a_l D & & & & q_{k'} a_{l'} D' \\
q_i a_j \arrow[ur] \arrow[r] \arrow[dr] & q_k' a_l' D' & \Rightarrow & q_i a_j \arrow[r] \arrow[dr] & \bar{q}_i a_j N \arrow[ur] \arrow[r] & q_k a_l D \\
 & q_k'' a_l'' D'' & & & q_k'' a_l'' D'' &
\end{tikzcd}$$

(это всё происходит логарифмически)

Строим переменные:

Имя	Смысл	Параметры (значение)
$Q[i,k]$ - булева переменная 0 или 1.	В момент времени i НМТ находится в состоянии q_k	$0 \le i \le p(n)$ $0 \le k \le m$
$H[i,j]$ " 0/1	В момент времени i НМТ осматривает ячейку с номером j	$0 \le i \le p(n)$ $-p(n) \le j \le p(n)$
$S[i,j,l]$ " 0/1	В момент времени i в j-ой ячейке ленты написан символ S_l	$0 \le i \le p(n)$ $-p(n) \le j \le p(n)$ $1 \le l \le \nu$
$C[i]$ ~ 0/1	В момент времени i машина выбрала верхнюю команду	$0 \le i \le p(n)$

$q_0, \underset{\uparrow q_{YES}}{q_1}, \ldots, q_m$ - состояния.

$S_1, S_2, \ldots, S_n$ - входной алфавит

Все числа, связанные с НМТ - это константы.

Клозы

Группа	Смысл	Клозы	Число
G_1	В каждый момент времени НМТ находится ровно в одном состоянии	$\forall i: Q[i,0] \vee Q[i,1] \vee \ldots \vee Q[i,m]$ $\forall i,k,k': \neg Q[i,k] \vee \neg Q[i,k']$ $k < k'$	$p(n)+1$ $(\forall i)$ $(p(n)+1) \cdot \frac{m(m+1)}{2}$

G_2	В каждой момент времени НМТ обозревает ровно одну ячейку.	$\forall i: H[i,-p(n))] \lor \ldots \lor H[i,p(n)]$ $\forall i,j,j': \neg H[i,j] \lor \neg H[i,j']$ $j < j'$	$p(n)+1$ $(p(n)+1) \cdot C^2_{2p(n)+1}$
G_3	В каждой момент времени в j-ой ячейке написан ровно один символ	$\forall i,j: S[i,j,1] \lor \ldots \lor S[i,j,\nu]$ $\forall i,j,l,l': l<l'$ $\neg S[i,j,l] \lor \neg S[i,j,l']$	$(p(n)+1)\cdot(2p(n)+1)$ $(p(n)+1)\, C^2_\nu$
G_4	Начальное состояние	$Q[0,0], H[0,0], S[0,j,l_j]$, где l_j — номер символа в j-ой ячейке в начальной момент времени.	$2p(n)+1$
G_5	Конечное состояние	$Q[p(n),1]$	1
G_6	Команды	$Q[i,k] \,\&\, H[i,j] \,\&\, S[i,j,l] \,\&$ $\&\, C[i] \to Q[i+1,k'] \,\&\, H[i+1,$ $j+D'] \,\&\, S[i+1,j,l']$ — выбираем верхнюю команду.	

$q_k S_l \to q_{k'} S_{l'} D'$ — верхняя команда

$q_k S_l \to q_{k''} S_{l''} D''$ — нижняя команда

		$Q[i,k] \,\&\, H[i,j] \,\&$ выбираем нижнюю $\&\, S[i,j,l] \,\&\, \neg C[i] \to$ $Q[i+1,k''] \,\&\, H[i+1,j+D''] \,\&$ $\&\, S[i+1,j,l'']$ $H[i,j] \,\&\, S[i,j',l] \to$ др. команд $S[i+1,j',l], j \neq j'$ НЕТ! Если МТ осматривает j ячейку, то в след. момент времени в другой ячейке ничего не изменится. $\neg H[i,j] \lor \neg S[i,j',l] \lor$ $\lor S[i+1,j',l]$	

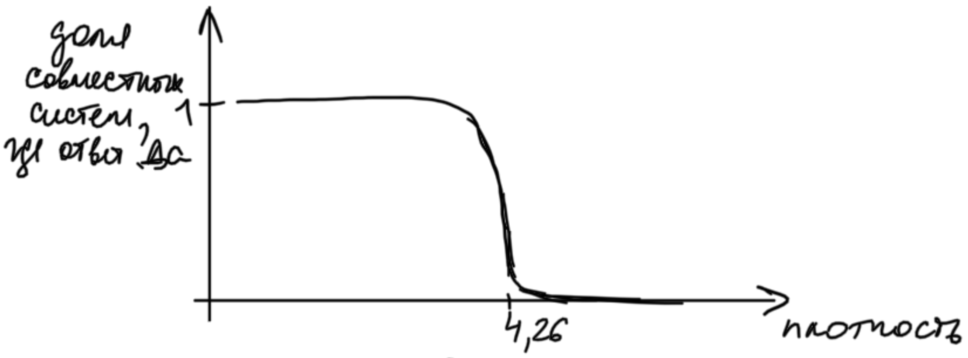
Придадим C[i] значения, какую команду выберем.

⇒ Если есть хорошие вычисления, то все эти клозы выполнены

⇐ Пусть можно подобрать значения переменных так, что клозы выполнятся, почему можно решить эту задачу?

Плотность SAT — отношение числа клозов к числу переменных.

Задачи с маленькой плотностью чаще всего имеют ответ Да, с большой — НЕТ.



! 2-SAT не NP-полна

Задача 3-SAT — о выполнимости, где в каждом клозе 3 литерала.

Лемма

3-SAT NP-полна

Достаточно просто свести SAT к 3-SAT

Надо рассмотреть 4 случая, для этого возьмём произвольный экземпляр задачи SAT, $x_1, \dots, x_m$ – переменные, $C_1, \dots, C_n$ – клозы.

Преобразуем каждый клоз из SAT в клоз с 3 литералами

1) в C_i один литерал ⇒ добавим 2 переменные z_1 и z_2 и составим 4 клоза:

$$C_i \vee z_1 \vee z_2,\ C_i \vee z_1 \vee \neg z_2,\ C_i \vee \neg z_1 \vee z_2,\ C_i \vee \neg z_1 \vee \neg z_2$$

Если $C_i = 1$, то все клозы 1

Если $c_i = 0$, то все клозы не могут быть равны 1 ⇒ все клозы $= 0$.

2) В c_i два литерала

$C_i \vee Z$, $C_i \vee \neg Z$

3) В c_i три литерала ⇒ всё хорошо.

4) В c_i более 3х литералов