Finite Model Theory / Descriptive Complexity:

Th: $\text{FO} \subseteq \mathbf{L} = \mathbf{DSPACE}[\log n]$

Fagin's Th: $\mathbf{NP} = \text{SO}\exists$.

$$\mathcal{A} \models \Phi \quad \Leftrightarrow \quad N(\text{bin}(\mathcal{A})) = 1$$

$$\Phi \quad = \quad (\exists C_0^{2k} \cdots C_{g-1}^{2k} \Delta^k)(\forall \bar{x})\psi$$

ψ is quantifier-free.

	Space 0	1	$\bar{s}$	$n-1$	n		n^k-1	Δ
Time 0	$\langle q_0, w_0 \rangle$	w_1	$\cdots$	w_{n-1}	$\sqcup$	$\cdots$	$\sqcup$	δ_0
1	w_0	$\langle q_1, w_1 \rangle$	$\cdots$	w_{n-1}	$\sqcup$	$\cdots$	$\sqcup$	δ_1
	$\vdots$	$\vdots$	$\vdots$			$\vdots$		$\vdots$
$\bar{t}$			a_{-1} a_0 a_1					δ_t
$\bar{t}+1$			b					δ_{t+1}
	$\vdots$	$\vdots$	$\vdots$			$\vdots$		$\vdots$
n^k-1	$\langle q_f, 1 \rangle$	$\sqcup$	$\cdots$	$\sqcup$	$\sqcup$	$\cdots$	$\sqcup$	

Accepting computation of N on input $w_0 w_1 \cdots w_{n-1}$

Theorem 19.1 [Cook-Levin Theorem]

SAT *is* **NP**-*complete.*

Proof: Let $B \in$ **NP** be arbitrary. By Fagin's theorem,

$$B \;=\; \{\mathcal{A} \mid \mathcal{A} \models \Phi\}$$

$$\Phi \;=\; (\exists C_0^{2k} \cdots C_{g-1}^{2k} \Delta^k)(\forall x_1 \cdots x_t)\psi(\bar{x})$$

with ψ quantifier-free and CNF: $\psi(\bar{x}) \;=\; \bigwedge_{j=1}^{r} T_j(\bar{x})$

with each T_j a disjunction of literals.

For all $\mathcal{A}$, have: $\mathcal{A} \in B \quad \Leftrightarrow \quad \mathcal{A} \models \Phi$

Want: $\mathcal{A} \in B \quad \Leftrightarrow \quad \varphi(\mathcal{A}) \in \mathrm{SAT}$

Let $\mathcal{A}$ be arbitrary, $\quad n = \|\mathcal{A}\|$

Wanted: $\quad \mathcal{A} \in B \quad \Leftrightarrow \quad \varphi(\mathcal{A}) \in \mathrm{SAT}$

Define: formula $\varphi(\mathcal{A})$ as follows:

boolean variables: $\quad C_i(e_1, \ldots, e_{2k}), \Delta(e_1, \ldots, e_k), \quad i = 0, \ldots, g-1, \quad e_1, \ldots, e_{2k} \in |\mathcal{A}|$

clauses: $\quad T'_j(\bar{e}), j = 1, \ldots, r, \quad \bar{e} \in |\mathcal{A}|^t$

$T'_j(\bar{e})$ is $T_j(\bar{e})$ with the following replacements:

$$\begin{array}{rcl} R(x_i, x_j) & \mapsto & \textbf{if } (\langle e_i, e_j \rangle \in R^{\mathcal{A}}) \textbf{ then } (\textbf{true}) \textbf{ else } (\textbf{false}) \\ x_i = x_j & \mapsto & \textbf{if } (e_1 = e_j) \textbf{ then } (\textbf{true}) \textbf{ else } (\textbf{false}) \\ x_i \leq x_j & \mapsto & \textbf{if } (e_1 \leq e_j) \textbf{ then } (\textbf{true}) \textbf{ else } (\textbf{false}) \\ C_i(x_{i_1}, \ldots x_{i_{2k}}) & \mapsto & C_i(e_{i_1}, \ldots e_{i_{2k}}) \\ \Delta(x_{i_1}, \ldots x_{i_k}) & \mapsto & \Delta(e_{i_1}, \ldots e_{i_k}) \end{array}$$

$$\Phi \equiv (\exists C_0^{2k} \cdots C_{g-1}^{2k} \Delta^k)(\forall x_1 \cdots x_t) \bigwedge_{j=1}^{r} T_j(\bar{x})$$

$$\varphi(\mathcal{A}) \equiv \bigwedge_{e_1, \ldots, e_t \in |\mathcal{A}|} \bigwedge_{j=1}^{r} T'_j(\bar{e})$$

$$\mathcal{A} \in B \quad \Leftrightarrow \quad \mathcal{A} \models \Phi \quad \Leftrightarrow \quad \varphi(\mathcal{A}) \in \mathrm{SAT} \spadesuit$$

Proposition 19.2

3-SAT $=$ $\{\varphi \in$ CNF-SAT $\mid \varphi$ *has* ≤ 3 *literals per clause*$\}$

3-SAT *is* **NP**-*complete.*

Proof: Show SAT $\leq$ 3-SAT.

Example:

$$C \quad = \quad (\ell_1 \vee \ell_2 \vee \cdots \vee \ell_7)$$

$$C' \equiv (\ell_1 \vee \ell_2 \vee d_1) \wedge (\overline{d_1} \vee \ell_3 \vee d_2) \wedge (\overline{d_2} \vee \ell_4 \vee d_3) \wedge (\overline{d_3} \vee \ell_5 \vee d_4) \wedge (\overline{d_4} \vee \ell_6 \vee \ell_7)$$

Claim: $C \in$ SAT $\Leftrightarrow$ $C' \in$ 3-SAT

In general do this construction for each clause independently.

$\varphi \in$ SAT $\Leftrightarrow$ $\varphi' \in$ 3-SAT ♠

What about reducing 3-SAT to SAT?

Can we do it?

Easily! The *identity function* serves as a reduction, because every 3-SAT instance is also a SAT instance with the same answer. This is an example of the general phenomenon of one problem being *a special case* of another. Another example was on HW#5, where LEVELLED-REACH was a special case of REACH and so clearly LEVELLED-REACH $\leq$ REACH.

But what does it prove to reduce 3-SAT **to** SAT**?**

Not much – only the fact that 3-SAT is in **NP** or that LEVELLED-REACH is in **NL**, neither of which was hard to prove anyway. To prove that a special case of a general problem is complete for some class, we have two options:

1. Reduce the general problem to the specific one, or
2. Show that the completeness proof for the general case can be adapted to always yield an instance of the special case

For example, in HW#5 the first method would be to follow my hint and reduce REACH to LEVELLED-REACH directly. The second method would be to show that when we map an arbitrary **NL** problem to a REACH instance, we can get a LEVELLED-REACH instance. (This happens if the TM in question keeps a clock on its worktape, for example.)

Proposition 19.3 3-COLOR *is* **NP**-*complete.*

Proof: Show 3-SAT $\leq$ 3-COLOR.

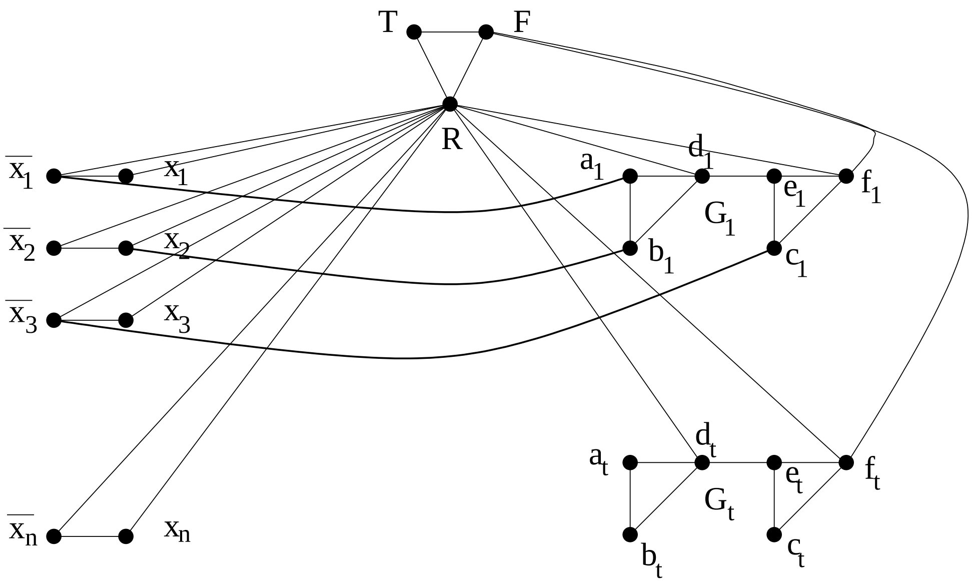
$$\varphi \quad = \quad C_1 \ \wedge \ C_2 \ \wedge \ \cdots \ \wedge \ C_t \quad \in \quad \text{3-CNF}$$

$$\text{VAR}(\varphi) \quad = \quad \{x_1, x_2, \ldots, x_n\}$$

Must build graph $G(\varphi)$ s.t.

$$\varphi \in \text{3-SAT} \qquad \Leftrightarrow \qquad G(\varphi) \in \text{3-COLOR}$$

Working assumption: 3-SAT requires $2^{\epsilon n}$ time.



G_1 encodes clause $C_1 = (\overline{x_1} \vee x_2 \vee \overline{x_3})$

Claim: Triangle a_1, b_1, d_1 serves as an "or"-gate:

d_1 may be colored "true" iff at least one of its inputs $\overline{x_1}, x_2$ is colored "true". Similarly, the output f_1 may be colored "true" iff at least one of d_1 and the third input, $\overline{x_3}$ is colored "true".

f_i can only be colored "true".

A three coloring of the literals can be extended to color G_i iff the corresponding truth assignment makes C_i true. ♠

Proposition 19.4 CLIQUE *is* **NP**-*complete.*

Proof:

Show SAT $\leq$ CLIQUE.

$$\varphi \quad = \quad C_1 \ \wedge \ C_2 \ \wedge \ \cdots \ \wedge \ C_t \quad \in \quad \text{CNF}$$

$$\text{VAR}(\varphi) \quad = \quad \{x_1, x_2, \ldots, x_n\}$$

Must build graph $g(\varphi)$ s.t.

$$\varphi \in \text{SAT} \qquad \Leftrightarrow \qquad g(\varphi) \in \text{CLIQUE}$$

$$L \ = \ \{x_1, \ldots, x_n, \overline{x}_1, \ldots, \overline{x}_n\}; \quad C \ = \ \{c_1, \ldots, c_t\}$$

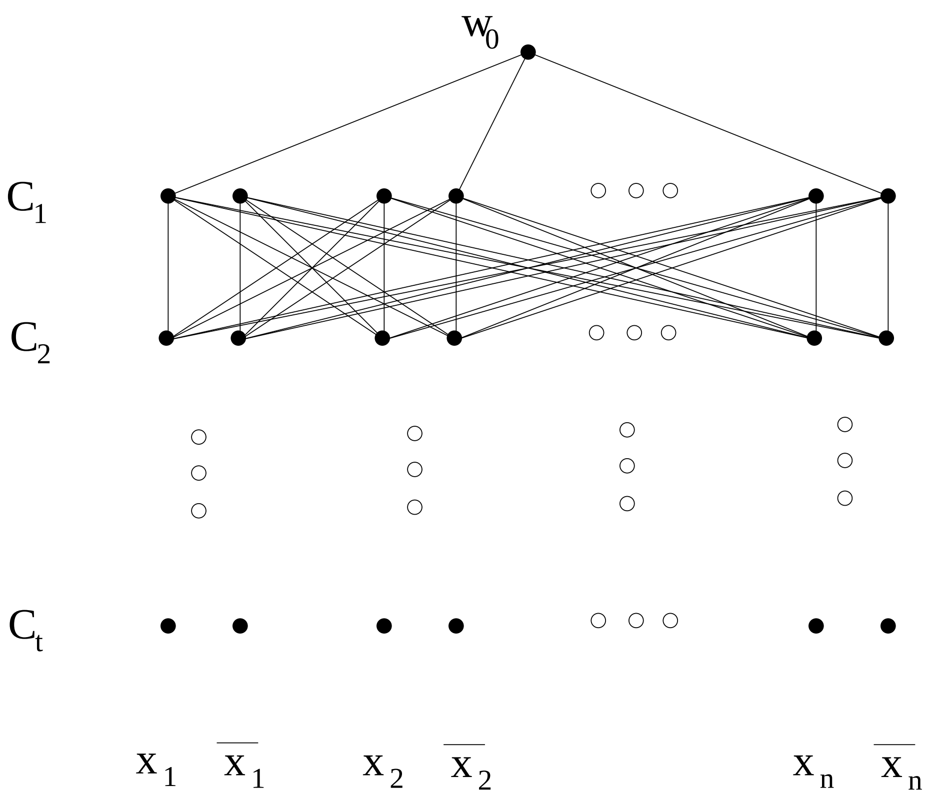
$$g(\varphi) \ = \ (V^{g(\varphi)}, E^{g(\varphi)}, k^{g(\varphi)})$$

$$V^{g(\varphi)} \ = \ (C \times L) \cup \{w_0\}$$

$$E^{g(\varphi)} \ = \ \{(\langle c_1, \ell_1\rangle, \langle c_2, \ell_2\rangle) \ | \ c_1 \neq c_2 \text{ and } \overline{\ell}_1 \neq \ell_2\} \ \cup$$

$$\{(w_0, \langle c, \ell\rangle), (\langle c, \ell\rangle, w_0) \ | \ \ell \text{ occurs in } c\}$$

$$k^{g(\varphi)} \ = \ t + 1$$



$$g(\varphi), \quad C_1 = (x_1 \vee \overline{x_2} \vee \overline{x_n})$$

$$\begin{aligned}
V^{g(\varphi)} &= (C \times L) \cup \{w_0\} \\
E^{g(\varphi)} &= \{(\langle c_1, \ell_1\rangle, \langle c_2, \ell_2\rangle) \mid c_1 \neq c_2 \text{ and } \overline{\ell}_1 \neq \ell_2\} \cup \\
&\quad \{(w_0, \langle c, \ell\rangle), (\langle c, \ell\rangle, w_0) \mid \ell \text{ occurs in } c\} \\
k^{g(\varphi)} &= t + 1
\end{aligned}$$

$$
\begin{aligned}
V^{g(\varphi)} &= (C \times L) \cup \{w_0\} \\
E^{g(\varphi)} &= \{(\langle c_1, \ell_1 \rangle, \langle c_2, \ell_2 \rangle) \mid c_1 \neq c_2 \text{ and } \overline{\ell}_1 \neq \ell_2\} \ \cup \\
&\quad \{(w_0, \langle c, \ell \rangle), (\langle c, \ell \rangle, w_0) \mid \ell \text{ occurs in } c\} \\
k^{g(\varphi)} &= t + 1
\end{aligned}
$$

$$
(\varphi \in \mathrm{SAT}) \quad \Leftrightarrow \quad (g(\varphi) \in \mathrm{CLIQUE})
$$

Claim: $g \in F(\mathbf{L})$

Proposition 19.5 *Subset Sum is* **NP**-*Complete.*

$$\{m_1, \ldots, m_r, T \in \mathbf{N} \mid (\exists S \subseteq \{1, \ldots, r\})(\sum_{i \in S} m_i = T)\}$$

Show 3-SAT $\leq$ Subset Sum.

$$\varphi \equiv C_1 \wedge C_2 \wedge \cdots \wedge C_t \quad \in \quad \text{3-CNF}$$
$$\text{VAR}(\varphi) = \{x_1, x_2, \ldots, x_n\}$$

Build $f \in F(\mathbf{L})$ such that for all φ,

$$\varphi \in \text{3-SAT} \qquad \Leftrightarrow \qquad f(\varphi) \in \text{Subset Sum}$$

	x_1	x_2	$\cdots$	x_n	C_1	C_2	$\cdots$	C_t	
T	1	1	$\cdots$	1	3	3	$\cdots$	3	
x_1	1	0	$\cdots$	0	1	0	$\cdots$	1	$C_1 = (x_1 \vee \overline{x_2} \vee x_3)$
$\overline{x_1}$	1	0	$\cdots$	0	0	1	$\cdots$	0	
x_2	0	1	$\cdots$	0	0	1	$\cdots$	1	$C_2 = (\overline{x_1} \vee x_2 \vee x_n)$
$\overline{x_2}$	0	1	$\cdots$	0	1	0	$\cdots$	0	
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$C_t = (x_1 \vee x_2 \vee \overline{x_n})$
x_n	0	0	$\cdots$	1	0	1	$\cdots$	0	
$\overline{x_n}$	0	0	$\cdots$	1	0	0	$\cdots$	1	
a_1	0	0	$\cdots$	0	1	0	$\cdots$	0	
b_1	0	0	$\cdots$	0	1	0	$\cdots$	0	
a_2	0	0	$\cdots$	0	0	1	$\cdots$	0	
b_2	0	0	$\cdots$	0	0	1	$\cdots$	0	
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	
a_t	0	0	$\cdots$	0	0	0	$\cdots$	1	
b_t	0	0	$\cdots$	0	0	0	$\cdots$	1	

Knapsack

Given n objects:

object	o_1 o_2 $\cdots$ o_n	
weight	w_1 w_2 $\cdots$ w_n	≥ 0
value	v_1 v_2 $\cdots$ v_n	

W = max weight I can carry in my knapsack.

Optimization Problem:

choose $S \subseteq \{1, \ldots, n\}$

to maximize $\sum_{i \in S} v_i$

such that $\sum_{i \in S} w_i \leq W$

Decision Problem:

Given $\bar{w}, \bar{v}, W, V$, can I get total value $\geq V$ while total weight is $\leq W$?

Proposition 19.6 *Knapsack is* **NP**-*Complete.*

Proof: Let $I = \langle m_1, \ldots m_n, T \rangle$ be an instance of Subset Sum.

Problem: $(\exists ? S \subseteq \{1, \ldots, n\})\big(\sum_{i \in S} m_i \; = \; T\big)$

Let $f(I) = \langle m_1, \ldots m_n, m_1, \ldots, m_n, T, T \rangle$ be an instance of Knapsack.

Claim: $I \in$ Subset Sum $\quad \Leftrightarrow \quad f(I) \in$ Knapsack

$$(\exists S \subseteq \{1, \ldots, n\})\big(\sum_{i \in S} m_i \; = \; T\big)$$

$$\Leftrightarrow$$

$$(\exists S \subseteq \{1, \ldots, n\})\big(\sum_{i \in S} m_i \; \geq \; T \quad \wedge \quad \sum_{i \in S} m_i \; \leq \; T\big) \spadesuit$$

Fact 19.7 *Even though Knapsack is* **NP**-*Complete there is an efficient dynamic programming algorithm that can closely approximate the maximum possible* V.