601 Lecture 18: Recall From Last Time

Savitch's Th: For $s(n) \geq \log n$,

$$\textbf{NSPACE}[s(n)] \quad \subseteq \quad \textbf{DSPACE}[(s(n))^2]$$

Immerman-Szelepcsényi Th: For $s(n) \geq \log n$,

$$\textbf{NSPACE}[s(n)] \quad = \quad \text{co-}\textbf{NSPACE}[s(n)]$$

Finite Model Theory

Consider the input (the object we are working on) to be a finite logical structure, e.g., a binary string, a graph, a relational database . . .

Def: FO is the set of first-order definable decision problems on finite structures. Let $S \subseteq \mathrm{STRUC_{fin}}[\Sigma]$.

$S \in \mathbf{FO}$ iff for some $\varphi \in \mathcal{L}(\Sigma)$

$$S = \{\mathcal{A} \in \mathrm{STRUC_{fin}}[\Sigma] \mid \mathcal{A} \models \varphi\}$$

FO is a complexity class:

the set of all first-order definable problems.

Addition

$$Q_+ : \mathrm{STRUC}[\Sigma_{AB}] \quad \longrightarrow \quad \mathrm{STRUC}[\Sigma_s]$$

$$\begin{array}{lccccccc} A & & a_1 & a_2 & \ldots & a_{n-1} & a_n \\ B & + & b_1 & b_2 & \ldots & b_{n-1} & b_n \\ \hline S & & s_1 & s_2 & \ldots & s_{n-1} & s_n \end{array}$$

$$C(i) \equiv \exists j > i \Big(A(j) \wedge B(j) \;\wedge\; (\forall k.j > k > i)(A(k) \vee B(k))\Big)$$

$$Q_+(i) \equiv A(i) \oplus B(i) \oplus C(i)$$

$$Q_+(k) \in \mathbf{FO}$$

Encode structure $\mathcal{A} \in \mathrm{STRUC}_{\mathrm{fin}}[\Sigma]$ as binary string: bin$(\mathcal{A})$.

Example:

- binary strings: bin$(\mathcal{A}_w) = w$
- graphs: $G = (\{1, \ldots, n\}, E, s, t)$
  bin$(G) = a_{11}a_{12} \ldots a_{nn}s_1s_2 \ldots s_{\log n}t_1 \ldots t_{\log n}$

Thm: **FO** $\subseteq$ **L** $=$ **DSPACE**$[\log n]$

Proof: Given: $\varphi \quad \equiv \quad \exists x_1 \forall x_2 \cdots \forall x_{2k} (\psi)$

Build **DSPACE**$[\log n]$ TM M s.t.,

$$\mathcal{A} \models \varphi \quad \Leftrightarrow \quad M(\text{bin}(\mathcal{A})) = 1$$

By induction on k.

Base case: $k = 0$.

$\varphi \equiv E(s,t)$

$\varphi \equiv s \leq t$

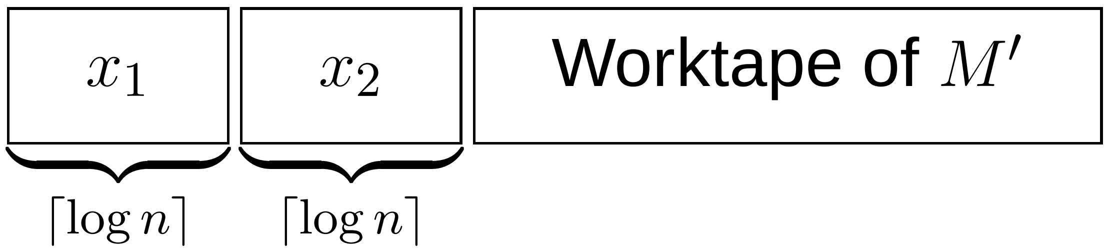
Inductive step: $\varphi \equiv \exists x_1 \forall x_2 (\varphi'); \quad \varphi' \equiv \exists x_3 \forall x_4 \cdots \forall x_{2k} (\psi)$

By inductive assumption, there is logspace TM M',

$$\mathcal{A} \models \varphi' \quad \Leftrightarrow \quad M'(\mathrm{bin}(\mathcal{A})) = 1$$

Modify M' by adding $2\lceil \log n \rceil$ worktape cells.

Worktape of M:



M cycles through all values of x_1 until it finds one such that for all x_2, M' accepts. □

Second-Order Logic

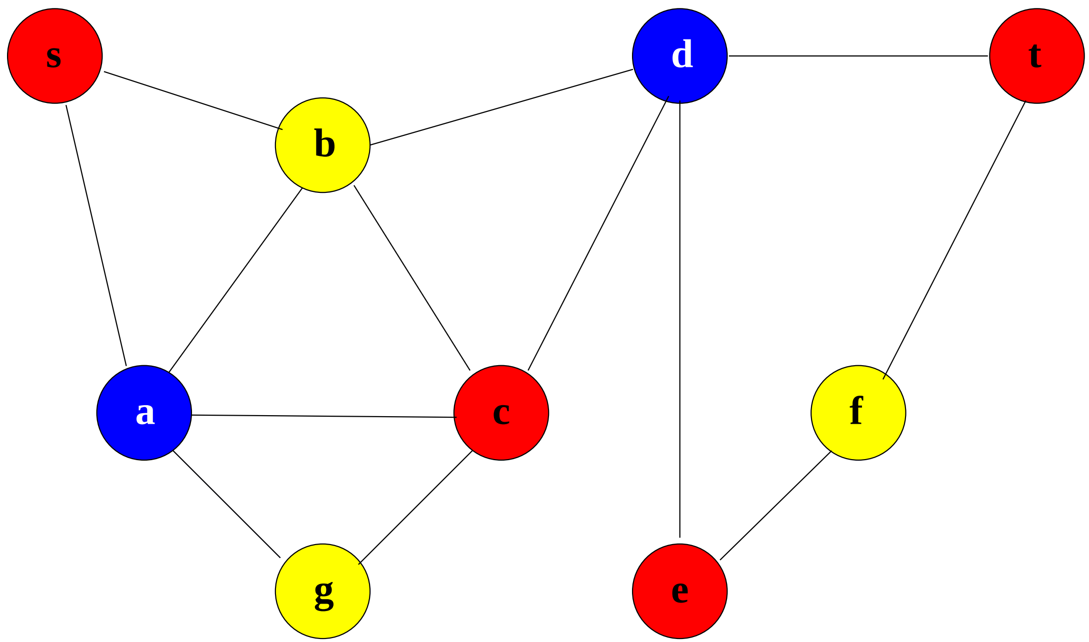
Second-order logic consists of first-order logic, plus new relation variables over which we may quantify.

$\exists A^r(\varphi)$: For some r-ary relation A, φ holds.

SO is the set of second-order expressible problems.

SO$\exists$ is the set of second-order existential problems.

$$\Phi_{\text{3-color}} \equiv \exists R^1 \exists Y^1 \exists B^1 \forall x \Big[\big(R(x) \vee Y(x) \vee B(x)\big)$$
$$\wedge \quad \forall y \Big(E(x,y) \rightarrow \neg\big(R(x) \wedge R(y)\big) \wedge$$
$$\neg\big(Y(x) \wedge Y(y)\big) \wedge$$
$$\neg\big(B(x) \wedge B(y)\big)\Big) \Big]$$

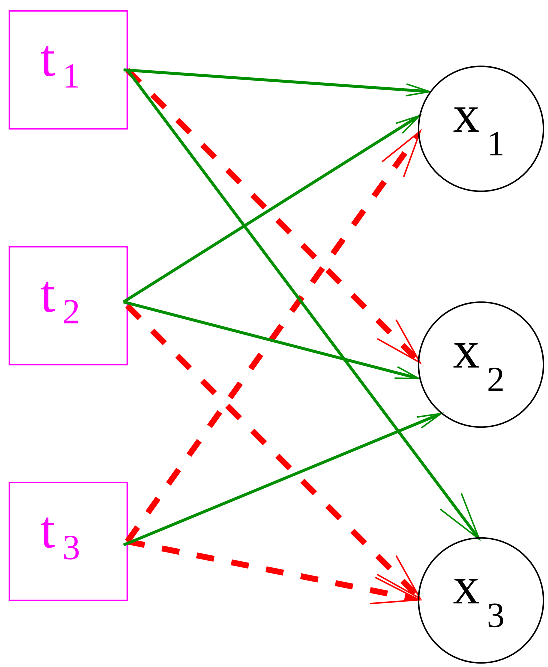


SAT is the set of boolean formulas in conjunctive normal form (CNF) that admit a satisfying assignment.

$$\Phi_{\mathrm{SAT}} \equiv \exists S^1 \forall t \exists x \, (C(t) \;\rightarrow\; (P(t,x) \wedge S(x)) \vee (N(t,x) \wedge \neg S(x)))$$

$$\begin{aligned} C(t) &\equiv \text{“}t \text{ is a clause; otherwise } t \text{ is a variable.”} \\ P(t,x) &\equiv \text{“Variable } x \text{ occurs positively in clause } t\text{.”} \\ N(t,x) &\equiv \text{“Variable } x \text{ occurs negatively in clause } t\text{.”} \end{aligned}$$

$$\begin{aligned} \varphi \;\equiv\; & (x_1 \vee \overline{x_2} \vee x_3) \quad \wedge \\ & (x_1 \vee x_2 \vee \overline{x_3}) \quad \wedge \\ & (\overline{x_1} \vee x_2 \vee \overline{x_3}) \end{aligned}$$

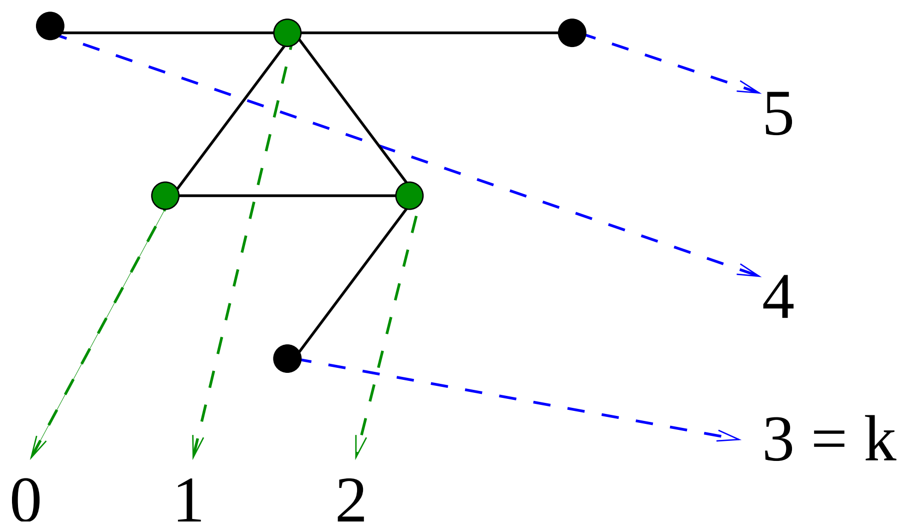


CLIQUE is the set of pairs $\langle G, k\rangle$ such that G is a graph having a complete subgraph of size k.

Let $\mathrm{Inj}(f)$ mean that f is an injective function, i.e., 1:1

$$\mathrm{Inj}(f) \quad \equiv \quad \forall xy\,(f(x) = f(y) \;\rightarrow\; x = y)$$

$$\Phi_{\text{CLIQUE}} \quad \equiv \quad \exists f^1.\mathrm{Inj}(f)\forall xy((x \neq y \wedge f(x) < k \wedge f(y) < k) \rightarrow E(x,y))$$



Fagin's Thm: **NP** $=$ **SO∃**.

Proof: NP $\supseteq$ **SO∃**:

Given **SO∃** sentence: $\Phi \;\equiv\; \exists R_1^{r_1} \ldots \exists R_k^{r_k}\, \psi \;\in\; \mathcal{L}(\Sigma)$

Build NP machine N s.t. for all $\mathcal{A} \in \mathrm{STRUC}_{\text{fin}}[\Sigma]$,

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

$\mathcal{A} \in \mathrm{STRUC}_{\text{fin}}[\Sigma]$, $n = \|\mathcal{A}\|$, $N(\text{bin}(\mathcal{A}))$ nondeterministically:

write binary string of length n^{r_1} representing R_1,
n^{r_2} representing R_2,
$\ldots$ $\ldots$,
n^{r_k} representing R_k.

$\mathcal{A}' = (\mathcal{A}, R_1, R_2, \ldots, R_k)$; N accepts iff $\mathcal{A}' \models \psi$.

FO $\subseteq$ **L** $\subseteq$ **NP** $\surd$

NP $\subseteq$ **SO∃**: Let N be an **NTIME[n^k]** TM.

To Write: **SO∃** sentence: $\Phi \equiv \exists C_0^{2k} \ldots C_{g-1}^{2k} \Delta^k (\varphi)$

meaning: "∃ accepting computation $\overline{C}, \Delta$ of N."

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

Fact: If have **numeric** relations and constants:

$\leq$,	Suc,	0,	*max*
ordering,	successor,	min elt.,	max elt.,

Then φ is universal: $\varphi \equiv \forall x_1 \cdots x_t (\alpha)$, α quantifier free

Encoding N's Computation

$$\text{Fix } \mathcal{A}, \quad n = \|\mathcal{A}\|$$

Possible contents of a computation cell for N:

$$\Gamma = \{\gamma_0, \ldots, \gamma_{g-1}\} = (Q \times \Sigma) \cup \Sigma$$

$C_i(s_1, \ldots, s_k, t_1, \ldots, t_k)$ means cell $\bar{s}$ at time $\bar{t}$ is symbol γ_i

$\Delta(\bar{t})$ means the $\bar{t} + 1^{\mathrm{st}}$ step of the computation makes choice "1"; otherwise it makes choice "0".

	Space							
	0	1	$\bar{s}$	$n-1$	n		n^k-1	Δ
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
Time	$\vdots$	$\vdots$	$\vdots$			$\vdots$		$\vdots$
$\bar{t}$			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 $w_0 w_1 \cdots w_{n-1}$

Write first-order sentence, $\varphi(\overline{C}, \Delta)$, saying that $\overline{C}, \Delta$ codes a valid accepting computation of N.

$$\varphi \quad \equiv \quad \alpha \wedge \beta \wedge \eta \wedge \zeta$$

$\alpha \quad \equiv \quad$ row 0 codes input bin$(\mathcal{A})$

$\beta \quad \equiv \quad \forall \bar{s}, \bar{t}, i \neq j \, (\neg(C_i(\bar{s}, \bar{t}) \wedge C_j(\bar{s}, \bar{t})))$

$\eta \quad \equiv \quad \forall \bar{t} \, (($row $\bar{t} + 1$ follows from row $\bar{t}$ via move $\Delta(\bar{t})$ of $N))$

$\zeta \quad \equiv \quad$ last row of computation is accept ID

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

$$\alpha \quad \equiv \quad \text{row 0 codes input bin}(\mathcal{A})$$

Assume Σ has only single unary relation symbol, R.

0	1		$n-1$	n		n^k-1
$\langle q_0, w_0\rangle$	w_1	$\cdots$	w_{n-1}	$\sqcup$	$\cdots$	$\sqcup$

$$\gamma_0 = 0;\ \gamma_1 = 1;\ \gamma_2 = \sqcup;\ \gamma_3 = \langle q_0, 0\rangle;\ \gamma_4 = \langle q_0, 1\rangle$$

$$\begin{aligned} \alpha \quad \equiv \quad & R(0) \to C_4(\bar{0}, \bar{0}) \\ \wedge \quad & \neg R(0) \to C_3(\bar{0}, \bar{0}) \\ \wedge \quad & \forall i > 0\, (R(i) \to C_1(\bar{0}i, \bar{0}) \\ & \qquad \wedge \neg R(i) \to C_0(\bar{0}i, \bar{0})) \\ \wedge \quad & \forall \bar{s} \geq n\, (C_2(\bar{s}, \bar{0})) \end{aligned}$$

Most interesting case: η

a_{-1}, a_0, a_1 leads to b via move δ of N: $\langle a_{-1}, a_0, a_1, \delta \rangle \xrightarrow{N} b$

$$\eta_1 \equiv \forall \bar{t} \,.\, \bar{t} < \overline{\mathit{max}} \;\; \forall \bar{s} \,.\, \bar{0} < \bar{s} < \overline{\mathit{max}} \bigwedge_{\langle a_{-1}, a_0, a_1, \delta \rangle \xrightarrow{N} b} \Big(\neg^{\delta} \Delta(\bar{t}) \vee$$

$$\neg C_{a_{-1}}(\bar{s} - 1, \bar{t}) \vee \neg C_{a_0}(\bar{s}, \bar{t}) \vee \neg C_{a_1}(\bar{s} + 1, \bar{t}) \vee C_b(\bar{s}, \bar{t} + 1) \Big)$$

Here $\neg^{\delta}$ is $\neg$ if $\delta = 1$ and it is the empty symbol if $\delta = 0$.

$$\eta \equiv \eta_0 \wedge \eta_1 \wedge \eta_2$$

where η_0 and η_2 encode the same information when $\bar{s} = \bar{0}$ and $\overline{\mathit{max}}$ respectively. $\square$

Cook's Thm: SAT is NP-complete.

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

$$B \;=\; \{\mathcal{A} \mid \mathcal{A} \models \Phi\}; \quad \Phi \;\equiv\; \exists C_0^{2k} \cdots C_{g-1}^{2k} \Delta^k \forall x_1 \cdots x_t \, (\alpha(\overline{x}))$$

with α quantifier-free and CNF,

$$\alpha(\bar{x}) \quad = \quad \bigwedge_{j=1}^{r} T_j(\bar{x})$$

with each T_j a disjunction of literals.

$\mathcal{A}$ arbitrary, $n = \|\mathcal{A}\|$, Define boolean formula $\varphi_{\mathcal{A}}$:

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

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

$T'_j(\bar{e})$ is $T_j(\bar{e})$ with $R(\bar{e})$, replaced by $\top$ or $\bot$ according as $\mathcal{A} \models R(\bar{e})$; $C_i(\bar{e})$, and $\Delta(\bar{e})$ are just boolean variables.

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

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

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

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

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

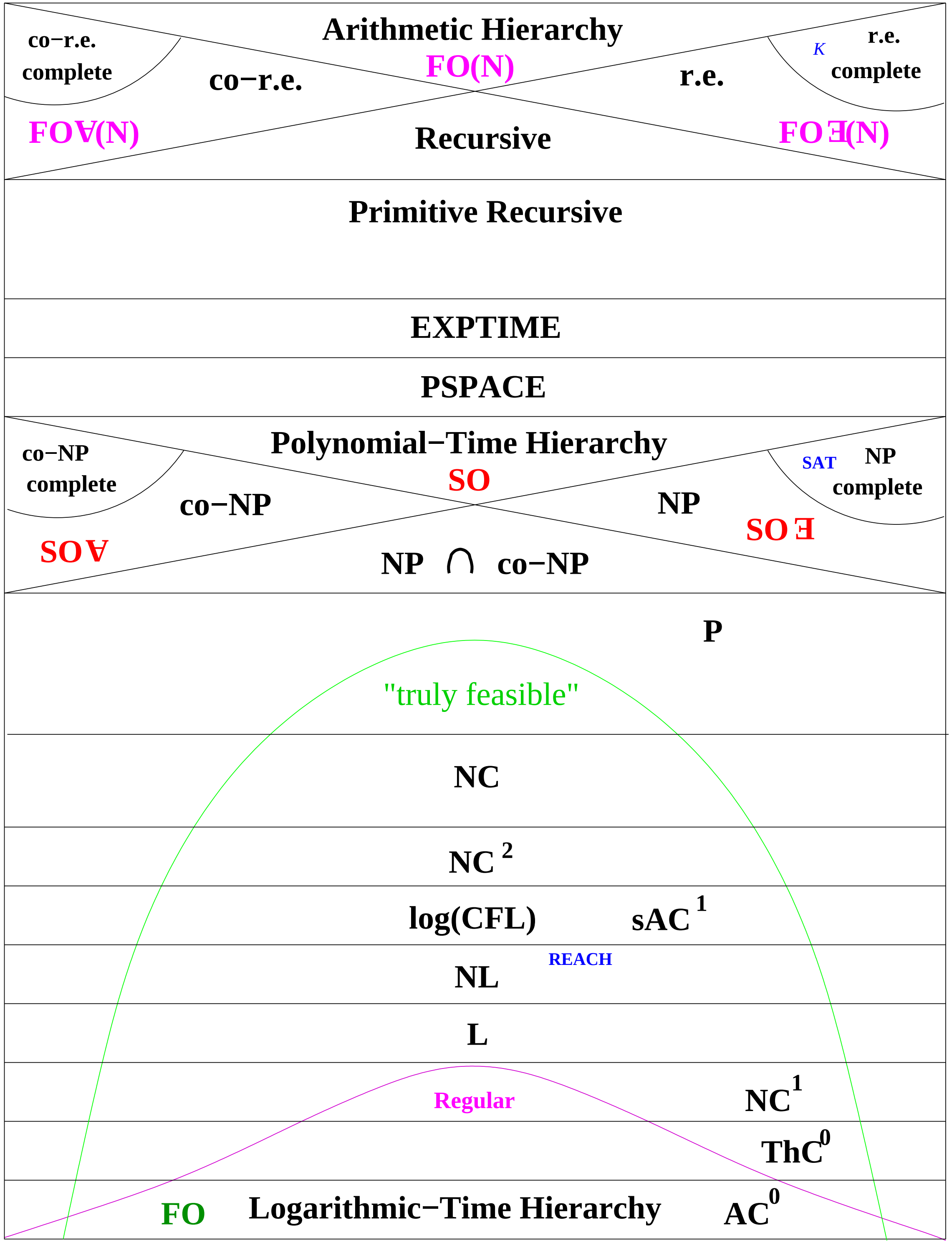
Proof: Show SAT $\leq$ 3-SAT.

Example: $C \equiv (\ell_1 \vee \ell_2 \vee \cdots \vee \ell_7)$

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

Claim: $C \in \text{SAT} \Leftrightarrow C' \in \text{3-SAT}$

Do this construction for each clause independently. □


Arithmetic Hierarchy
FO(N)
co−r.e. complete
co−r.e.
r.e.
K
r.e. complete
FO∀(N)
Recursive
FO∃(N)
Primitive Recursive
EXPTIME
PSPACE
Polynomial−Time Hierarchy
SO
co−NP complete
co−NP
NP
SAT
NP complete
SO∃
SO∀
NP ∩ co−NP
P
"truly feasible"
NC
NC 2
log(CFL)
sAC 1
REACH
NL
L
Regular
NC1
ThC0
FO
Logarithmic−Time Hierarchy
AC0