Finite Model Theory and Descriptive Complexity

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 . . .

Definition 6.1 FO is the set of first-order definable decision problems on finite structures. □

Let $S \subseteq \mathrm{STRUC}_{\mathrm{fin}}[\Sigma]$.

$$S \in \mathrm{FO} \quad \text{iff} \quad \text{for some } \varphi \in \mathcal{L}(\Sigma) \quad S = \{\mathcal{A} \in \mathrm{STRUC}_{\mathrm{fin}}[\Sigma] \mid \mathcal{A} \models \varphi\}$$

FO is a complexity class: the set of all first-order definable decision problems.

Addition $\quad Q_+ : \text{STRUC}[\Sigma_{AB}] \rightarrow \text{STRUC}[\Sigma_s]$

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

$$\begin{aligned} C(i) &\equiv \exists j > i \Big(A(j) \wedge B(j) \ \wedge \ (\forall k . j > k > i)(A(k) \vee B(k)) \Big) \\ Q_+(i) &\equiv A(i) \oplus B(i) \oplus C(i) \\ Q_+(k) &\in \text{FO} \end{aligned}$$

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

Example:

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

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.,

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

By induction on k.

Base case: $k = 0$.

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

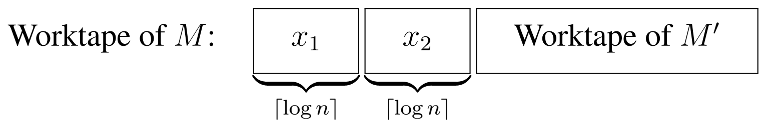
$\varphi \equiv s \leq t$

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',

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

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



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

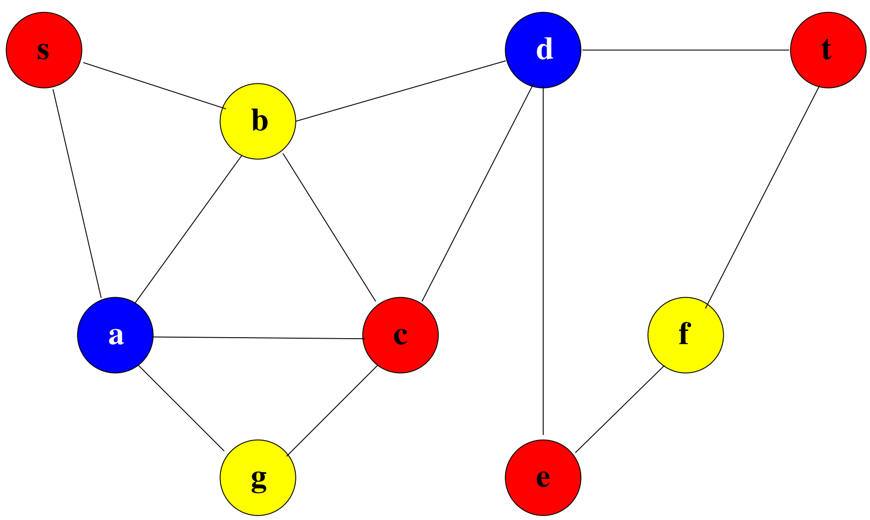
Second-Order Logic, consists of first-order logic, plus new relation variables over which we may quantify.

$$\exists A^r(\varphi): \quad \text{For some } r\text{-ary relation } A,\ \varphi \text{ holds.}$$

SO is the set of second-order expressible problems.

SO∃ is the set of second-order existential problems.

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

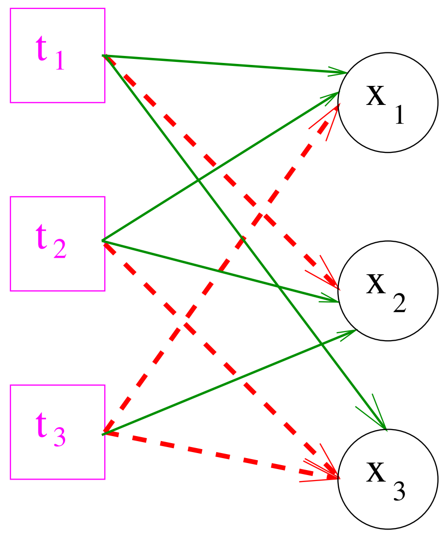


SAT is the set of boolean formulas in conjunctive normal form (CNF) that admit a satisfying assignment.

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

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

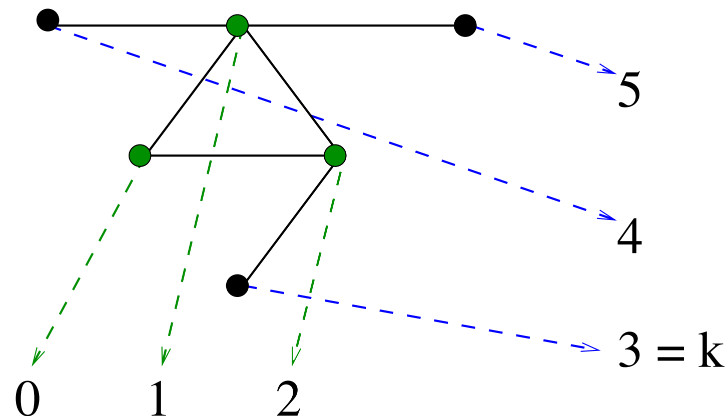
$$\begin{aligned} \varphi \;\equiv\; & (x_1 \vee \overline{x_2} \vee x_3) \quad \wedge \\ & (x_1 \vee x_2 \vee \overline{x_3}) \quad \wedge \\ & (\overline{x_1} \vee x_2 \vee \overline{x_3}) \end{aligned}$$



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

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

$$\begin{aligned}\text{Inj}(f) &\equiv \forall xy\,(f(x) = f(y) \;\rightarrow\; x = y)\\ \Phi_{\text{CLIQUE}} &\equiv \exists f^1.\text{Inj}(f)\forall xy((x \neq y \wedge f(x) < k \wedge f(y) < k) \rightarrow E(x,y))\end{aligned}$$



Fagin's Thm: NP $=$ SO$\exists$.

Proof: NP $\supseteq$ SO$\exists$:

Given SO$\exists$ 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 $\mathcal{A} \in \text{STRUC}_{\text{fin}}[\Sigma]$,

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

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

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

$$\text{FO} \subseteq \text{L} \;\subseteq\; \text{NP} \qquad \surd$$

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

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

meaning: "$\exists$ 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,	min,	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 $\bar{t} + 1^{\text{st}}$ 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.

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

$$\begin{aligned}
\alpha &\equiv \text{row 0 codes input bin}(\mathcal{A}) \\
\beta &\equiv \forall \bar{s}, \bar{t}, i \neq j\, (\neg(C_i(\bar{s}, \bar{t}) \wedge C_j(\bar{s}, \bar{t}))) \\
\eta &\equiv \forall \bar{t}\, ((\text{row } \bar{t} + 1 \text{ follows from row } \bar{t} \text{ via move } \Delta(\bar{t}) \text{ of } N)) \\
\zeta &\equiv \text{last row of computation is accept ID}
\end{aligned}$$

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

$$\begin{aligned}
\Phi &\equiv \exists C_0^{2k} C_1^{2k} \cdots C_{g-1}^{2k} \Delta^k (\varphi) \\
&\equiv \text{“}\exists \text{ an accepting compution: } N(\text{me}) = 1\text{”}
\end{aligned}$$

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

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}	$\sqcup$	$\cdots$	$\sqcup$

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

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

Most interesting case: η

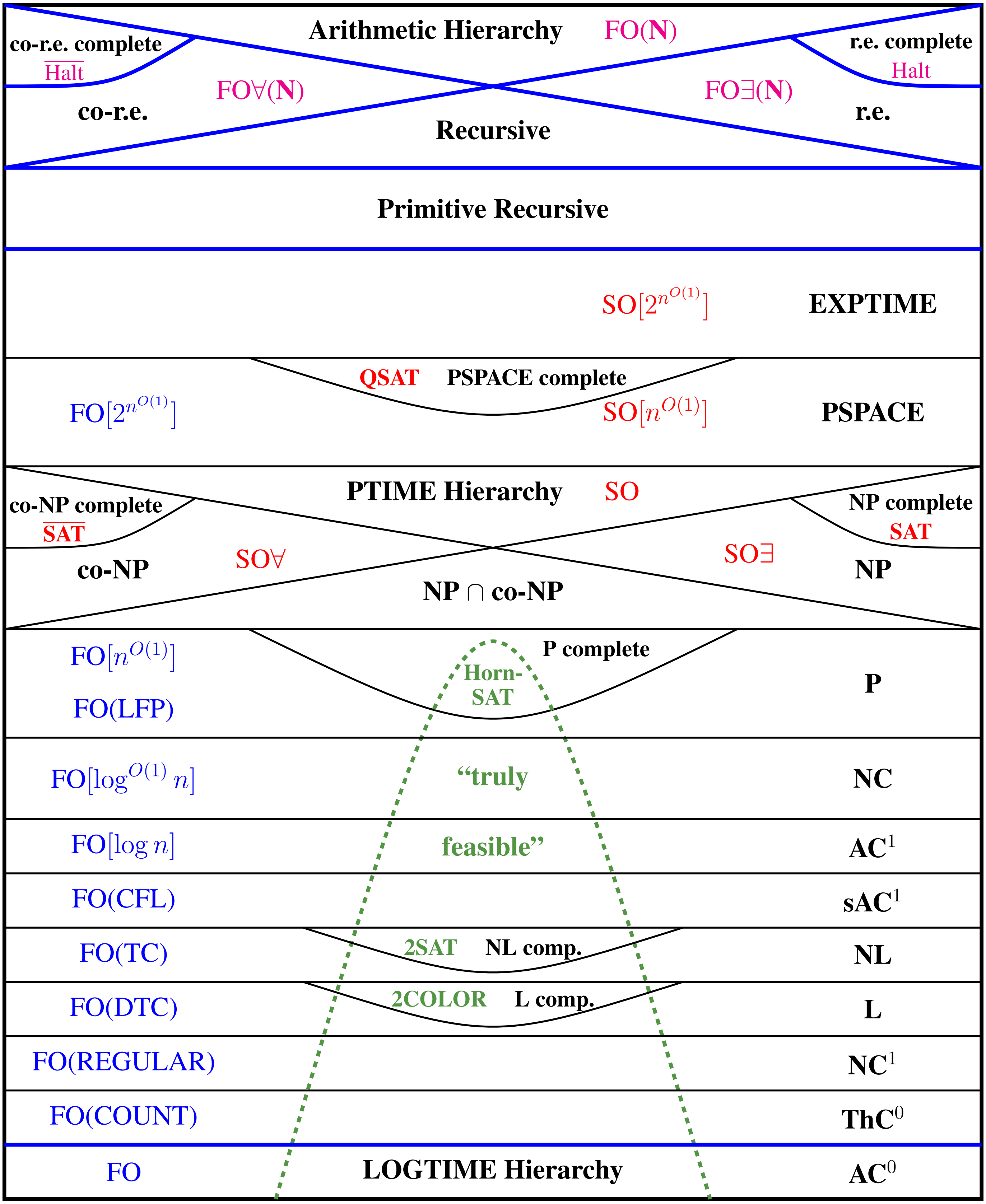
a_{-1}, a_0, a_1 leads to b via move δ of N: $\langle a_{-1}, a_0, a_1, \delta\rangle \stackrel{N}{\to} b$

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

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 $\bar{s} = \overline{0}$ and $\overline{max}$ respectively. $\square$


Arithmetic Hierarchy
FO(**N**)
co-r.e. complete
$\overline{\text{Halt}}$
r.e. complete
Halt
FO∀(**N**)
FO∃(**N**)
co-r.e.
r.e.
Recursive
Primitive Recursive
$\text{SO}[2^{n^{O(1)}}]$
EXPTIME
QSAT
PSPACE complete
$\text{FO}[2^{n^{O(1)}}]$
$\text{SO}[n^{O(1)}]$
PSPACE
PTIME Hierarchy
SO
co-NP complete
$\overline{\text{SAT}}$
NP complete
SAT
SO∀
SO∃
co-NP
NP
NP ∩ co-NP
$\text{FO}[n^{O(1)}]$
P complete
Horn-SAT
P
FO(LFP)
$\text{FO}[\log^{O(1)} n]$
“truly
NC
$\text{FO}[\log n]$
feasible”
AC^1
FO(CFL)
sAC^1
FO(TC)
2SAT
NL comp.
NL
FO(DTC)
2COLOR
L comp.
L
FO(REGULAR)
NC^1
FO(COUNT)
ThC^0
FO
LOGTIME Hierarchy
AC^0