Descriptive Complexity

Neil Immerman

www.cs.umass.edu/∼immerman

Please ask questions during my talk because two-way communication is more fun and allows much more understanding!

Descriptive Complexity

Descriptive Complexity

$$\begin{array}{c}\textbf{Input}\\ q_1\ q_2\ \cdots\ q_n\end{array} \mapsto \boxed{\textbf{Computation}} \mapsto \begin{array}{c}\textbf{Output}\\ a_1\ a_2\ \cdots\ a_i\ \cdots\ a_{n^k}\end{array}$$

Descriptive Complexity

Input **Output**

$$q_1\ q_2\ \cdots\ q_n \quad \longmapsto \quad \boxed{\textbf{Computation}} \quad \longmapsto \quad a_1\ a_2\ \cdots\ \boxed{a_i}\ \cdots\ a_{n^k}$$

Descriptive Complexity

Input

$q_1 \; q_2 \; \cdots \; q_n$

$$\longmapsto \boxed{\textbf{Computation}} \longmapsto$$

Output

a

Descriptive Complexity

$$\begin{array}{ccccc} \textbf{Input} & & & & \textbf{Output} \\ & \mapsto & \boxed{\textbf{Computation}} & \mapsto & \\ q_1\ q_2\ \cdots\ q_n & & & & a \end{array}$$

How much computation is needed to **check** if input has property S ?

Descriptive Complexity

Input $\longmapsto$ $\boxed{\textbf{Computation}}$ $\longmapsto$ **Output**

$q_1\ q_2\ \cdots\ q_n$ $\qquad\qquad$ a

How much computation is needed to **check** if input has property S ?

How rich a language do we need to **express** property S ?

Encode Input as Finite Logical Structure

Graph $G = (\{v_1, \ldots, v_n\}, E, s, t)$

Binary String

$$\mathcal{A}_w = (\{p_1, \ldots, p_8\}, S)$$

$$S = \{p_2, p_5, p_7, p_8\}$$

$$w = 01001011$$

Vocabularies: $\tau_g = (E^2, s, t)$, $\tau_s = (S^1)$

First-Order Logic

input symbols: from τ

variables: $x, y, z, \ldots$

boolean connectives: $\wedge, \vee, \neg$

quantifiers: $\forall, \exists$

numeric symbols: $=, \leq, +, \times, 0, \mathit{max}$

$$\alpha \equiv \forall x \exists y (E(x, y)) \in \mathcal{L}(\tau_g)$$

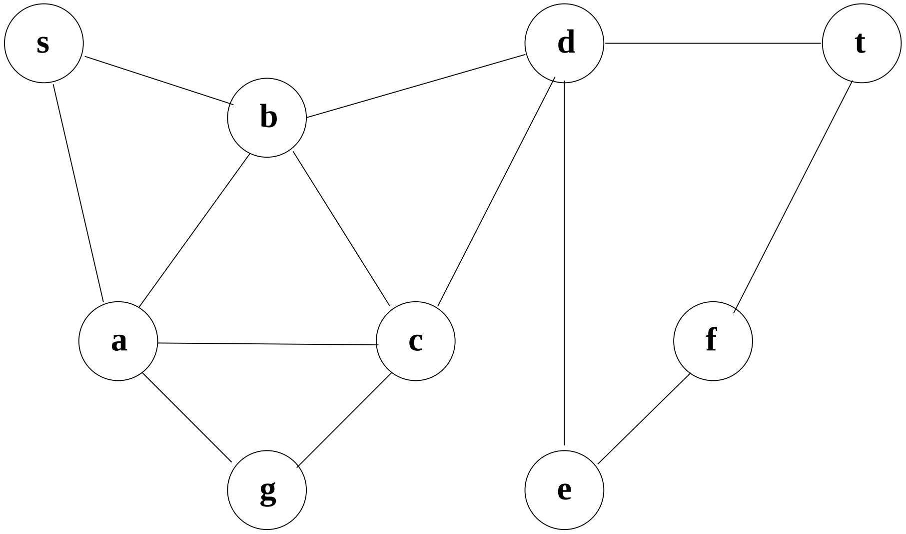
$$\beta \equiv \exists x \forall y (x \leq y \wedge S(x)) \in \mathcal{L}(\tau_s)$$

$$\beta \equiv S(0) \in \mathcal{L}(\tau_s)$$

Second-Order Logic

$=$ first-order logic $+$ quantifiable relational variables

$$\begin{aligned}\Phi_{3-\mathrm{color}} \;\equiv\; & \exists\, R^1\, Y^1\, B^1\, \forall\, x\, y\, ((R(x) \vee Y(x) \vee B(x)) \;\wedge \\ & (E(x,y) \to (\neg(R(x) \wedge R(y)) \wedge \neg(Y(x) \wedge Y(y)) \\ & \qquad \wedge \quad \neg(B(x) \wedge B(y)))))\end{aligned}$$



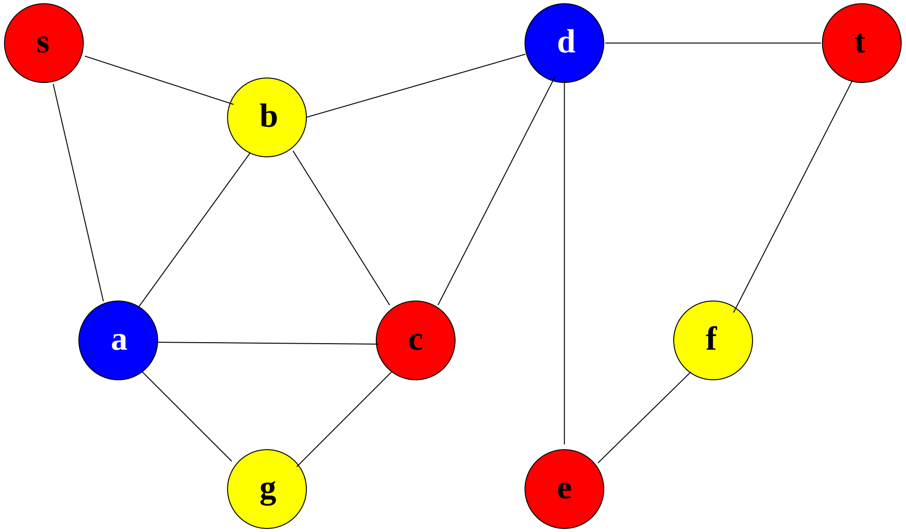
Second-Order Logic

Fagin's Theorem: NP $=$ SO$\exists$

$$\Phi_{3-\text{color}} \equiv \exists R^1 Y^1 B^1 \forall x\, y\, ((R(x) \vee Y(x) \vee B(x)) \wedge$$

$$(E(x,y) \rightarrow (\neg(R(x) \wedge R(y)) \wedge \neg(Y(x) \wedge Y(y))$$

$$\wedge \quad \neg(B(x) \wedge B(y)))))$$



Addition is First-Order

$$Q_+ : \text{STRUC}[\tau_{AB}] \longrightarrow \text{STRUC}[\tau_s]$$

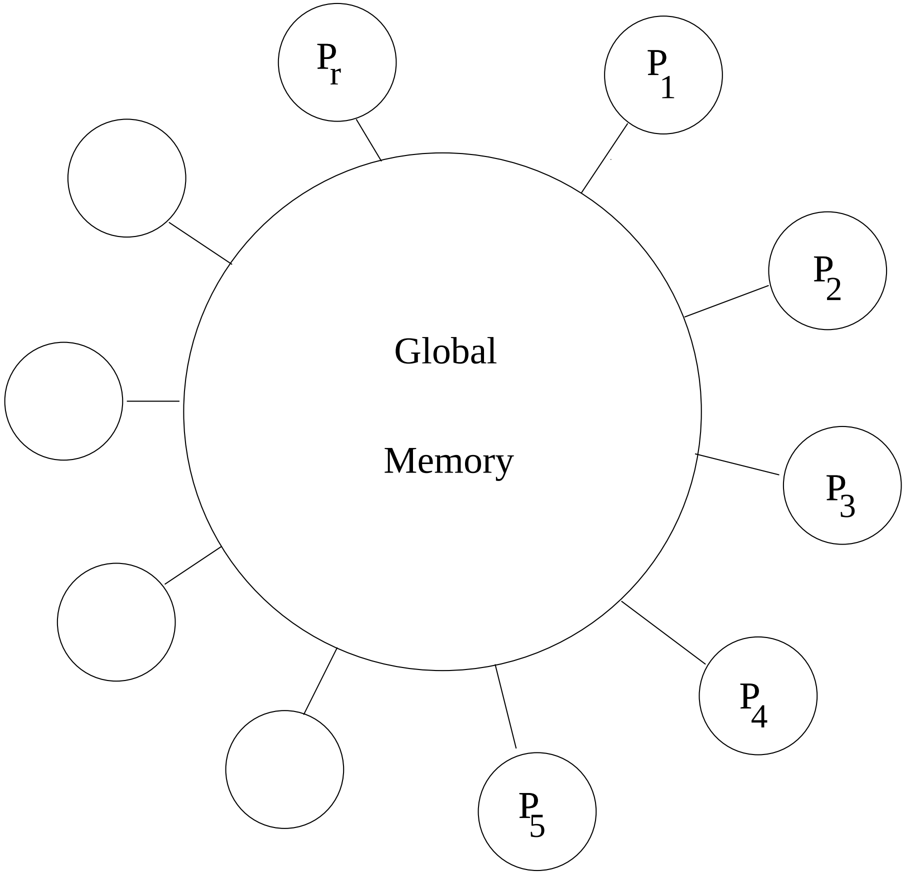
$$\begin{array}{llcccccc}
A & & a_1 & a_2 & \dots & a_{n-1} & a_n \\
B & + & b_1 & b_2 & \dots & b_{n-1} & b_n \\
\hline
S & & s_1 & s_2 & \dots & s_{n-1} & s_n
\end{array}$$

$$\begin{aligned}
C(i) &\equiv (\exists j > i)\Big(A(j) \wedge B(j) \;\wedge \\
&\qquad\qquad (\forall k . j > k > i)(A(k) \vee B(k))\Big)
\end{aligned}$$

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

Parallel Machines

CRAM$[t(n)]$ $=$ CRCW-PRAM-TIME$[t(n)]$-HARD$[n^{O(1)}]$

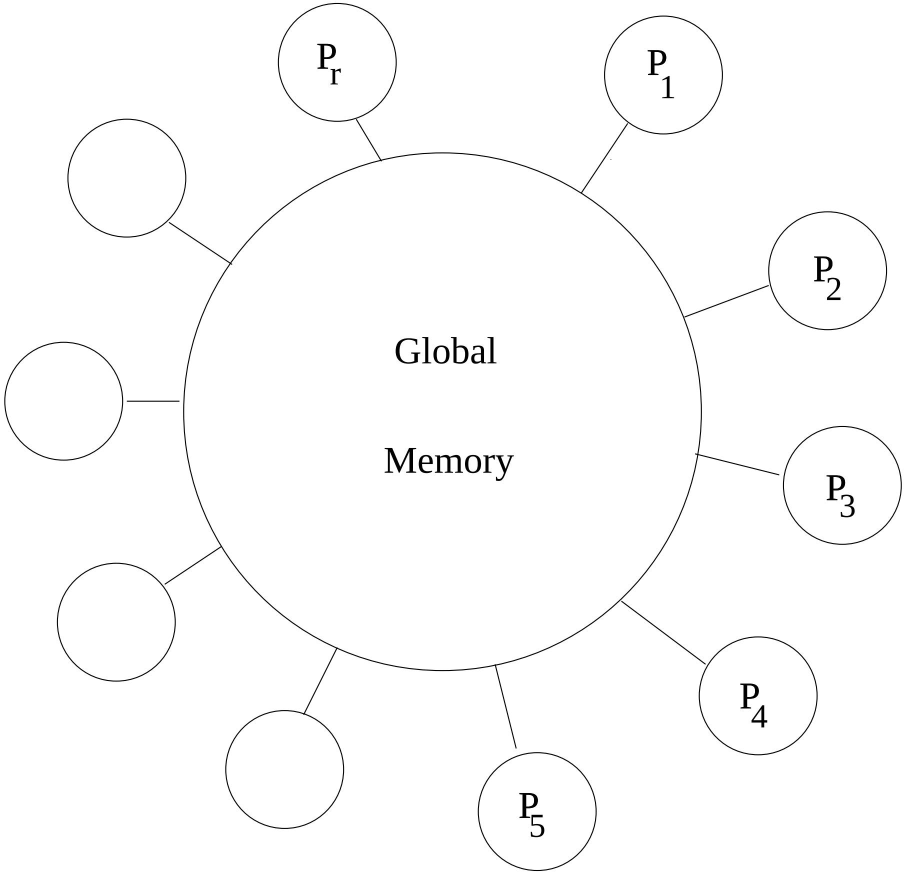


synchronous, concurrent read and write

$n^{O(1)}$ processors and memory

Parallel Machines

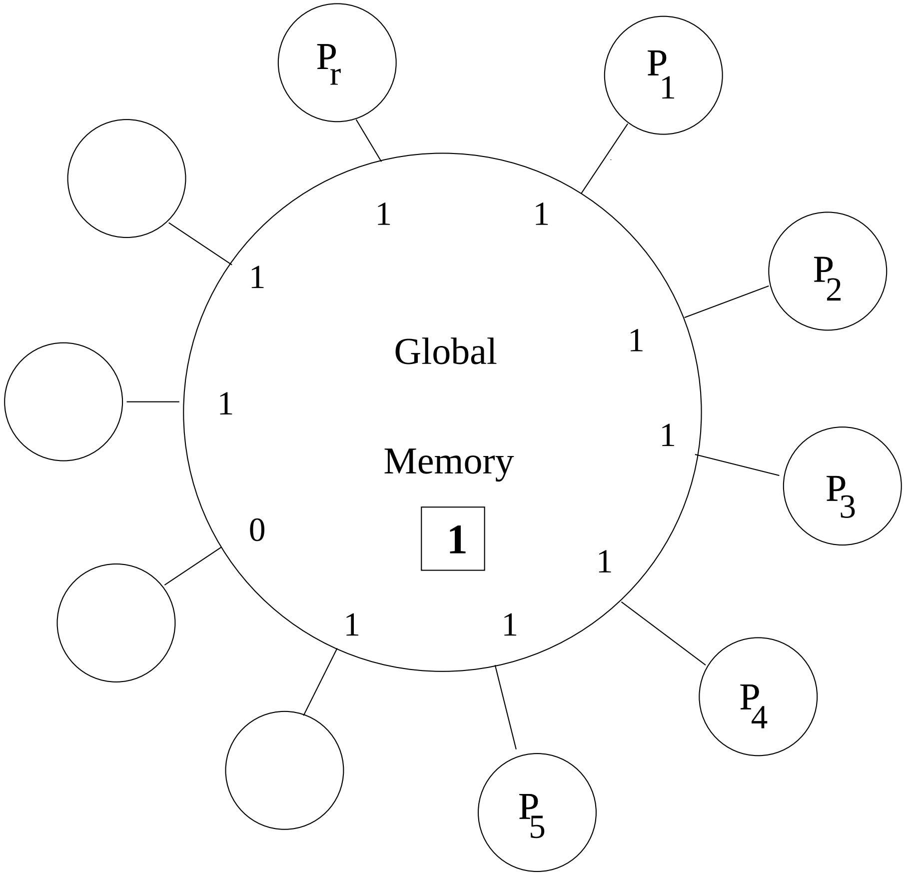
CRAM$[t(n)]$ $=$ CRCW-PRAM-TIME$[t(n)]$-HARD$[n^{O(1)}]$



Quantifiers are parallel: If array $A[x] : x = 1, \ldots, r$ in memory, $\forall x(A(x)) \;\equiv\;$ **write**(1); proc p_i: **if** $(A[i] = 0)$ **then** **write**(0)

Parallel Machines

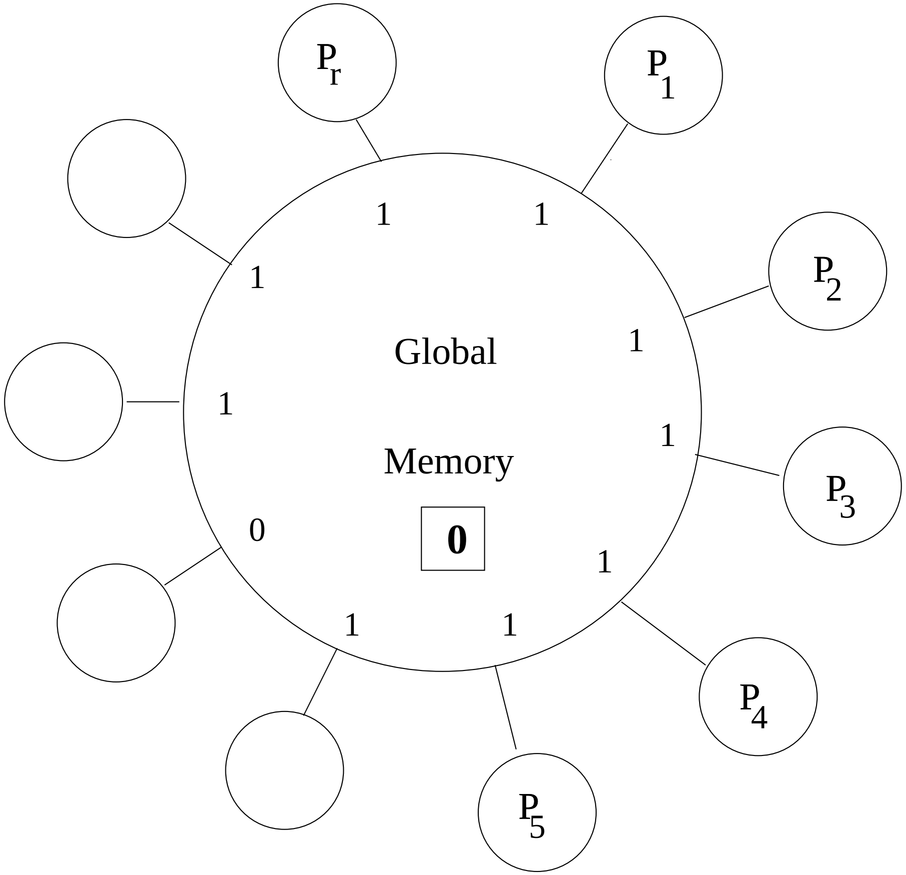
CRAM$[t(n)]$ $=$ CRCW-PRAM-TIME$[t(n)]$-HARD$[n^{O(1)}]$



Quantifiers are parallel: If array $A[x] : x = 1, \ldots, r$ in memory, $\forall x(A(x)) \;\equiv\;$ **write**(1); proc p_i: **if** $(A[i] = 0)$ **then** **write**(0)

Parallel Machines

CRAM$[t(n)]$ $=$ CRCW-PRAM-TIME$[t(n)]$-HARD$[n^{O(1)}]$



Quantifiers are parallel: If array $A[x] : x = 1, \ldots, r$ in memory, $\forall x(A(x)) \;\equiv\;$ **write**(1); proc p_i: **if** $(A[i] = 0)$ **then** **write**(0)

Inductive Definitions

$$E^\star(x,y) \quad \equiv \quad x = y \ \vee \ E(x,y) \ \vee \ \exists z(E^\star(x,z) \wedge E^\star(z,y))$$

Inductive Definitions

$$E^\star(x,y) \quad \equiv \quad x = y \ \vee \ E(x,y) \ \vee \ \exists z(E^\star(x,z) \wedge E^\star(z,y))$$

$$\varphi_{tc}(R,x,y) \quad \equiv \quad x = y \ \vee \ E(x,y) \ \vee \ \exists z(R(x,z) \wedge R(z,y))$$

$$\mathcal{A} \in \text{REACH} \quad \Leftrightarrow \quad \mathcal{A} \models (\mathbf{LFP}\varphi_{tc})(s,t)$$

Inductive Definitions

$$E^\star(x,y) \quad \equiv \quad x = y \;\vee\; E(x,y) \;\vee\; \exists z(E^\star(x,z) \wedge E^\star(z,y))$$

$$\varphi_{tc}(R,x,y) \quad \equiv \quad x = y \;\vee\; E(x,y) \;\vee\; \exists z(R(x,z) \wedge R(z,y))$$

$$\mathcal{A} \in \mathrm{REACH} \quad \Leftrightarrow \quad \mathcal{A} \models (\mathbf{LFP}\varphi_{tc})(s,t)$$

Thus, $\mathrm{REACH} \in \mathbf{IND}[\log n]$.

Inductive Definitions

$$E^\star(x,y) \quad \equiv \quad x = y \ \vee \ E(x,y) \ \vee \ \exists z(E^\star(x,z) \wedge E^\star(z,y))$$

$$\varphi_{tc}(R,x,y) \quad \equiv \quad x = y \ \vee \ E(x,y) \ \vee \ \exists z(R(x,z) \wedge R(z,y))$$

$$\mathcal{A} \in \mathrm{REACH} \quad \Leftrightarrow \quad \mathcal{A} \models (\mathbf{LFP}\varphi_{tc})(s,t)$$

Thus, $\mathrm{REACH} \in \mathbf{IND}[\log n]$.

Next, we'll show that $\mathrm{REACH} \in \mathbf{FO}[\log n]$.

$$\varphi_{tc}(R,x,y) \quad \equiv \quad x = y \ \vee \ E(x,y) \ \vee \ (\exists z)(R(x,z) \wedge R(z,y))$$

1. Dummy universal quantification for base case:

$$\begin{aligned}\varphi_{tc}(R,x,y) &\equiv (\forall z.M_1)(\exists z)(R(x,z) \wedge R(z,y)) \\ M_1 &\equiv \neg(x = y \vee E(x,y))\end{aligned}$$

$$\varphi_{tc}(R,x,y) \quad \equiv \quad x = y \;\vee\; E(x,y) \;\vee\; (\exists z)(R(x,z) \wedge R(z,y))$$

1. Dummy universal quantification for base case:

$$\begin{aligned} \varphi_{tc}(R,x,y) &\equiv (\forall z.M_1)(\exists z)(R(x,z) \wedge R(z,y)) \\ M_1 &\equiv \neg(x = y \vee E(x,y)) \end{aligned}$$

2. Using $\forall$, replace two occurrences of R with one:

$$\begin{aligned} \varphi_{tc}(R,x,y) &\equiv (\forall z.M_1)(\exists z)(\forall uv.M_2)R(u,v) \\ M_2 &\equiv (u = x \wedge v = z) \;\vee\; (u = z \wedge v = y) \end{aligned}$$

$$\varphi_{tc}(R,x,y) \quad \equiv \quad x=y \ \vee \ E(x,y) \ \vee \ (\exists z)(R(x,z)\wedge R(z,y))$$

1. Dummy universal quantification for base case:

$$\begin{aligned}\varphi_{tc}(R,x,y) &\equiv (\forall z.M_1)(\exists z)(R(x,z)\wedge R(z,y))\\ M_1 &\equiv \neg(x=y\vee E(x,y))\end{aligned}$$

2. Using $\forall$, replace two occurrences of R with one:

$$\begin{aligned}\varphi_{tc}(R,x,y) &\equiv (\forall z.M_1)(\exists z)(\forall uv.M_2)R(u,v)\\ M_2 &\equiv (u=x\wedge v=z) \ \vee \ (u=z\wedge v=y)\end{aligned}$$

3. Requantify x and y.

$$M_3 \quad \equiv \quad (x=u\wedge y=v)$$

$$\varphi_{tc}(R,x,y) \equiv (\forall z.M_1)(\exists z)(\forall uv.M_2)(\exists xy.M_3)R(x,y)$$

Define the quantifier block:

$$\mathrm{QB}_{tc} \;\equiv\; (\forall z.M_1)(\exists z)(\forall uv.M_2)(\forall xy.M_3)$$

Operator φ_{tc} is equivalent to QB_{tc}:

$$\varphi_{tc}(R,x,y) \quad \equiv \quad [\mathrm{QB}_{tc}]R(x,y)$$

Thus, for any r, $\qquad \varphi_{tc}^{r}(\emptyset) \;\equiv\; [\mathrm{QB}_{tc}]^{r}(\mathbf{false})$

Thus, for any structure $\mathcal{A} \in \mathrm{STRUC}[\tau_g]$,

$$\begin{aligned} \mathcal{A} \in \mathrm{REACH} \quad &\Leftrightarrow \quad \mathcal{A} \models (\mathbf{LFP}\varphi_{tc})(s,t) \\ &\Leftrightarrow \quad \mathcal{A} \models ([\mathrm{QB}_{tc}]^{\lceil 1+\log \|\mathcal{A}\| \rceil}\,\mathbf{false})(s,t) \end{aligned}$$

$\mathbf{CRAM}[t(n)] \ = \ $ concurrent parallel random access machine; polynomial hardware, parallel time $O(t(n))$

$\mathbf{IND}[t(n)] \ = \ $ first-order, depth $t(n)$ inductive definitions

$\mathbf{FO}[t(n)] \ = \ t(n)$ repetitions of a block of restricted quantifiers:

$$\mathrm{QB} = [(Q_1 x_1.M_1) \cdots (Q_k x_k.M_k)]; \quad M_i \text{ quantifier-free}$$

$$\varphi_n \quad = \quad \underbrace{[\mathrm{QB}][\mathrm{QB}] \cdots [\mathrm{QB}]}_{t(n)} M_0$$

parallel time = inductive depth = quantifier-block iteration

Thm: For all constructible, polynomially bounded $t(n)$,

$$\mathbf{CRAM}[t(n)] \;=\; \mathbf{IND}[t(n)] \;=\; \mathbf{FO}[t(n)]$$

Thm: For all $t(n)$, even beyond polynomial,

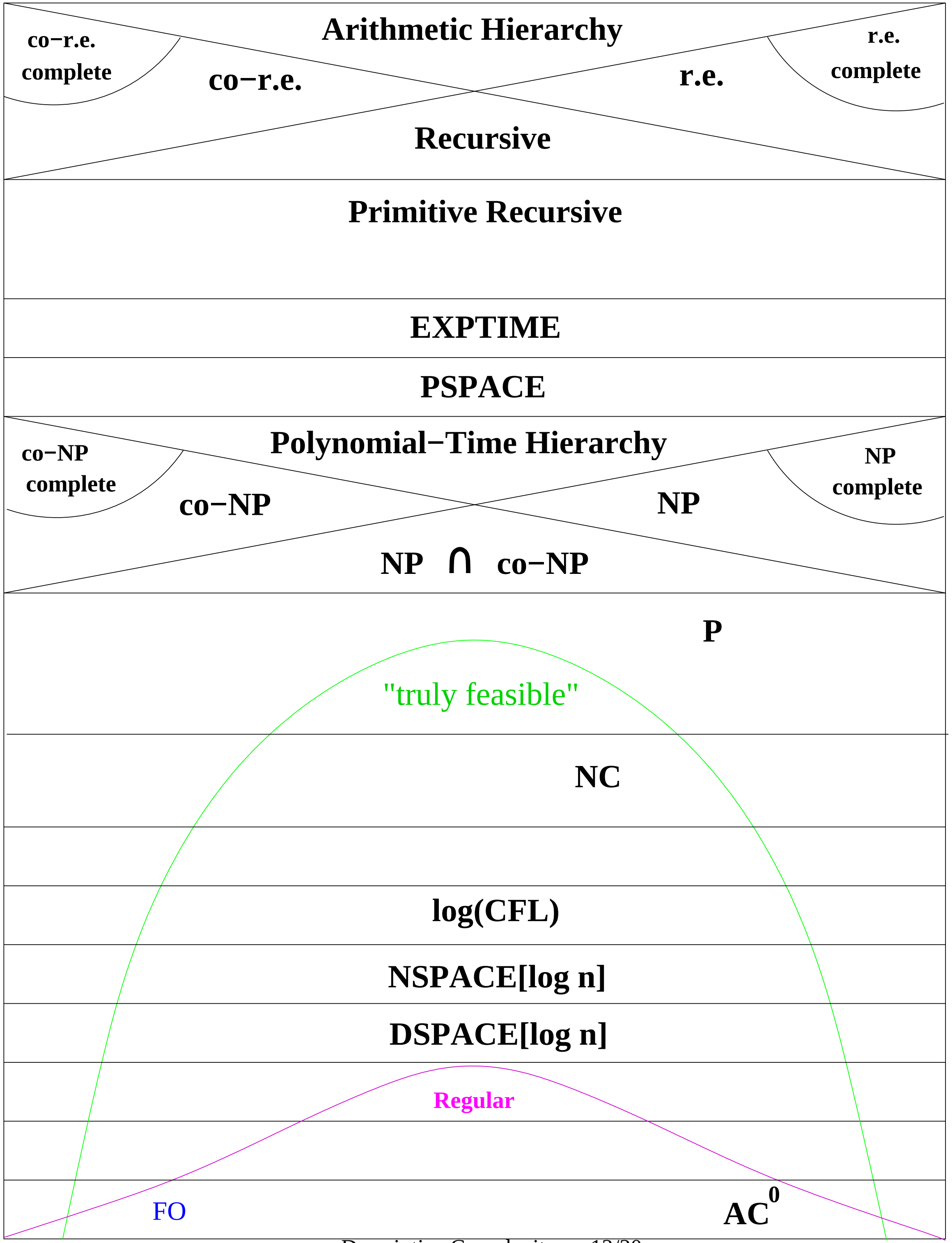
$$\mathbf{CRAM}[t(n)] \quad = \quad \mathbf{FO}[t(n)]$$

Thm: For $v = 1, 2, \ldots,$ $\quad \mathbf{DSPACE}[n^v] \; = \; \mathbf{VAR}[v+1]$

Number of variables corresponds to amount of hardware.

Since variables range over a universe of size n, a constant number of variables can specify a polynomial number of gates:

A bounded number of variables corresponds to polynomially much hardware.



Arithmetic Hierarchy

co−r.e. complete

co−r.e.

r.e.

r.e. complete

Recursive

Primitive Recursive

EXPTIME

PSPACE

Polynomial−Time Hierarchy

co−NP complete

co−NP

NP

NP complete

NP ∩ co−NP

P

"truly feasible"

NC

log(CFL)

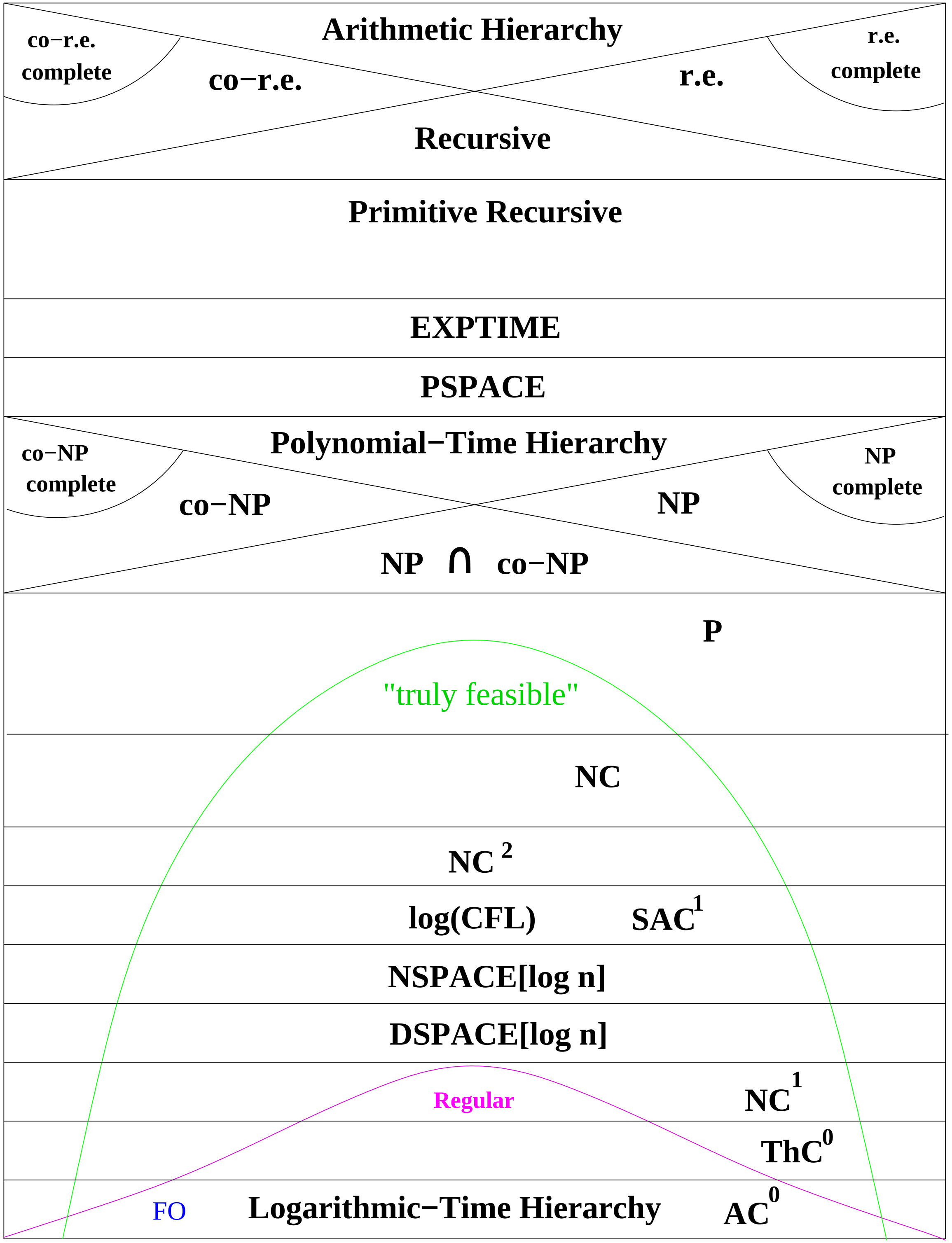
NSPACE[log n]

DSPACE[log n]

Regular

FO

AC^0



Arithmetic Hierarchy

co−r.e. complete

co−r.e.

r.e.

r.e. complete

Recursive

Primitive Recursive

EXPTIME

PSPACE

Polynomial−Time Hierarchy

co−NP complete

co−NP

NP

NP complete

NP ∩ co−NP

P

"truly feasible"

NC

NC^2

log(CFL) SAC^1

NSPACE[log n]

DSPACE[log n]

Regular

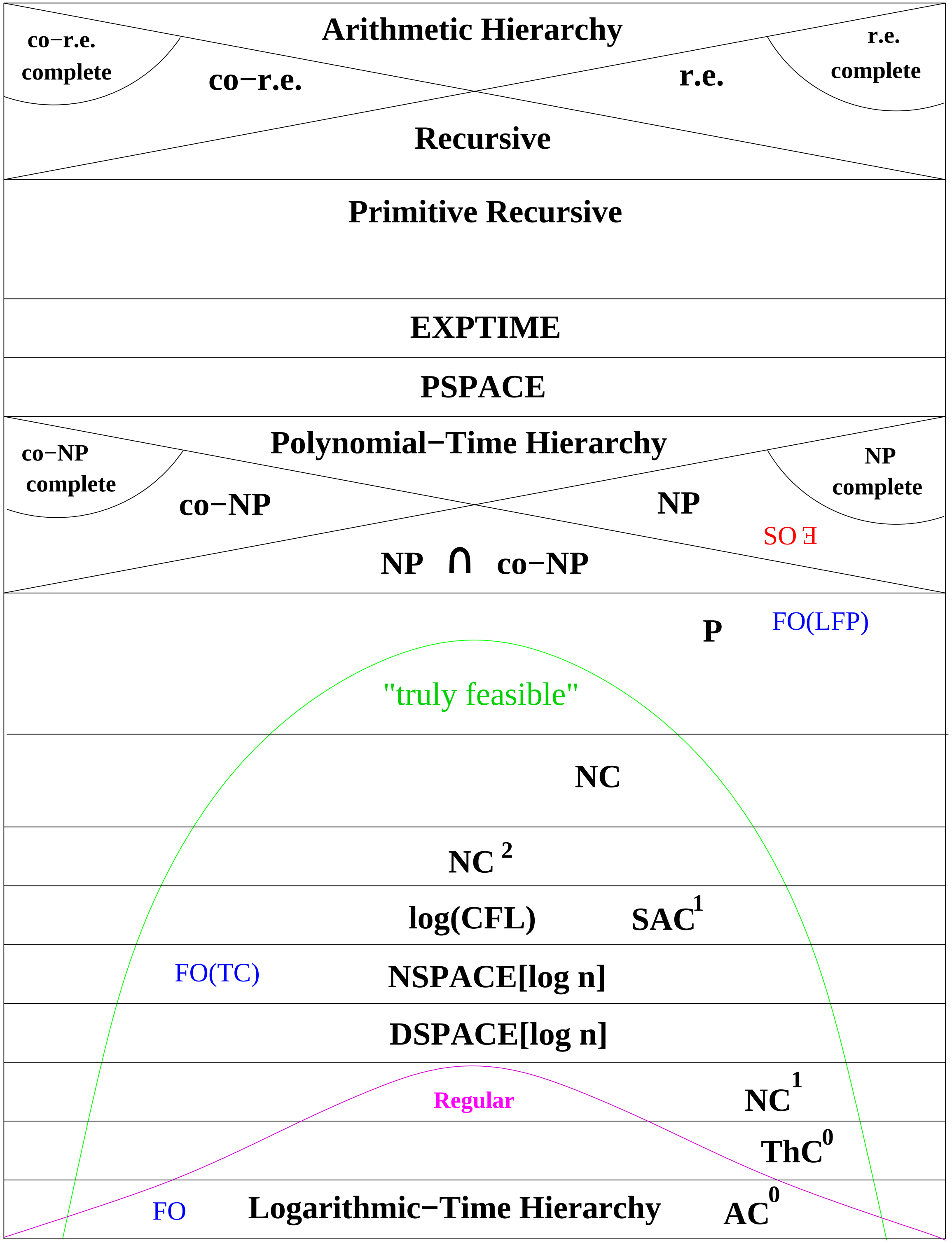
NC^1

ThC^0

FO

Logarithmic−Time Hierarchy

AC^0



Arithmetic Hierarchy

co−r.e. complete

co−r.e.

r.e.

r.e. complete

Recursive

Primitive Recursive

EXPTIME

PSPACE

Polynomial−Time Hierarchy

co−NP complete

co−NP

NP

NP complete

SO∃

NP ∩ co−NP

P

FO(LFP)

"truly feasible"

NC

NC^2

log(CFL)

SAC^1

FO(TC)

NSPACE[log n]

DSPACE[log n]

Regular

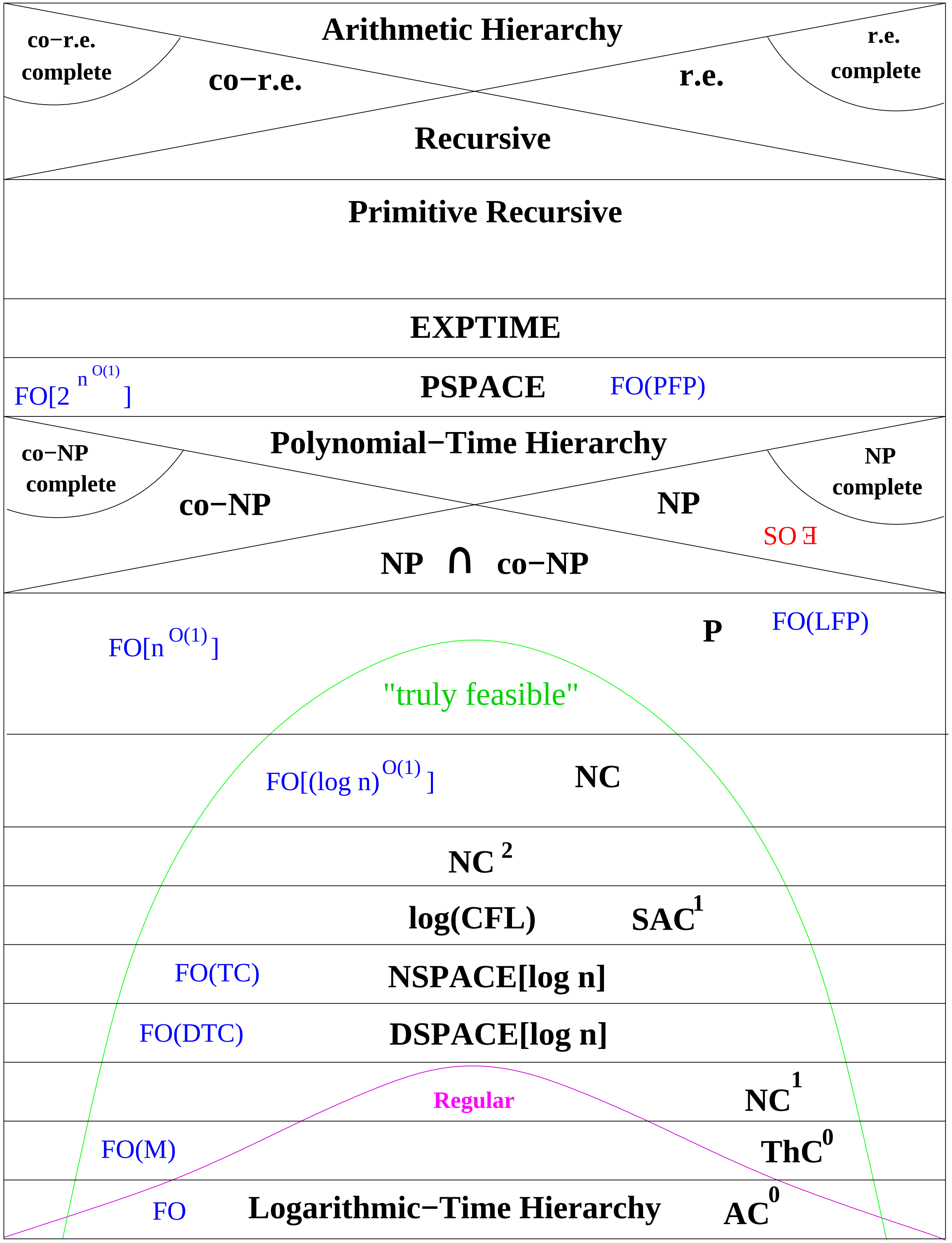
NC^1

ThC^0

FO

Logarithmic−Time Hierarchy

AC^0



Arithmetic Hierarchy

co−r.e. complete

co−r.e.

r.e.

r.e. complete

Recursive

Primitive Recursive

EXPTIME

FO[$2^{n^{O(1)}}$]

PSPACE

FO(PFP)

Polynomial−Time Hierarchy

co−NP complete

co−NP

NP

NP complete

SO∃

NP ∩ co−NP

FO[$n^{O(1)}$]

P

FO(LFP)

"truly feasible"

FO[$(\log n)^{O(1)}$]

NC

NC^2

log(CFL)

SAC^1

FO(TC)

NSPACE[log n]

FO(DTC)

DSPACE[log n]

Regular

NC^1

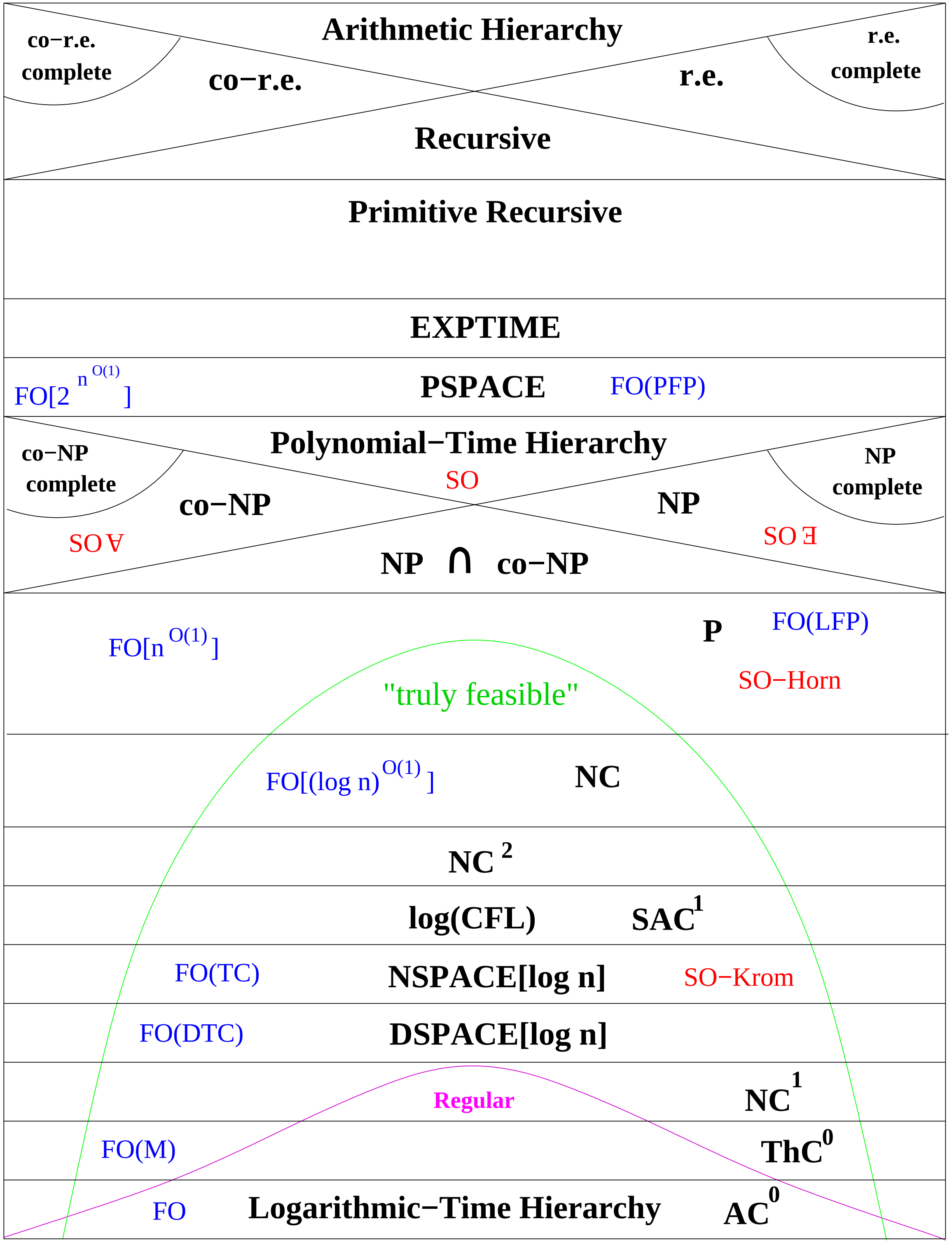
FO(M)

ThC^0

FO

Logarithmic−Time Hierarchy

AC^0



Arithmetic Hierarchy

co−r.e. complete

co−r.e.

r.e.

r.e. complete

Recursive

Primitive Recursive

EXPTIME

FO[$2^{n^{O(1)}}$]

PSPACE

FO(PFP)

Polynomial−Time Hierarchy

co−NP complete

SO

co−NP

NP

NP complete

SO ∀

SO ∃

NP ∩ co−NP

FO[$n^{O(1)}$]

P

FO(LFP)

"truly feasible"

SO−Horn

FO[$(\log n)^{O(1)}$]

NC

NC^2

log(CFL)

SAC^1

FO(TC)

NSPACE[log n]

SO−Krom

FO(DTC)

DSPACE[log n]

Regular

NC^1

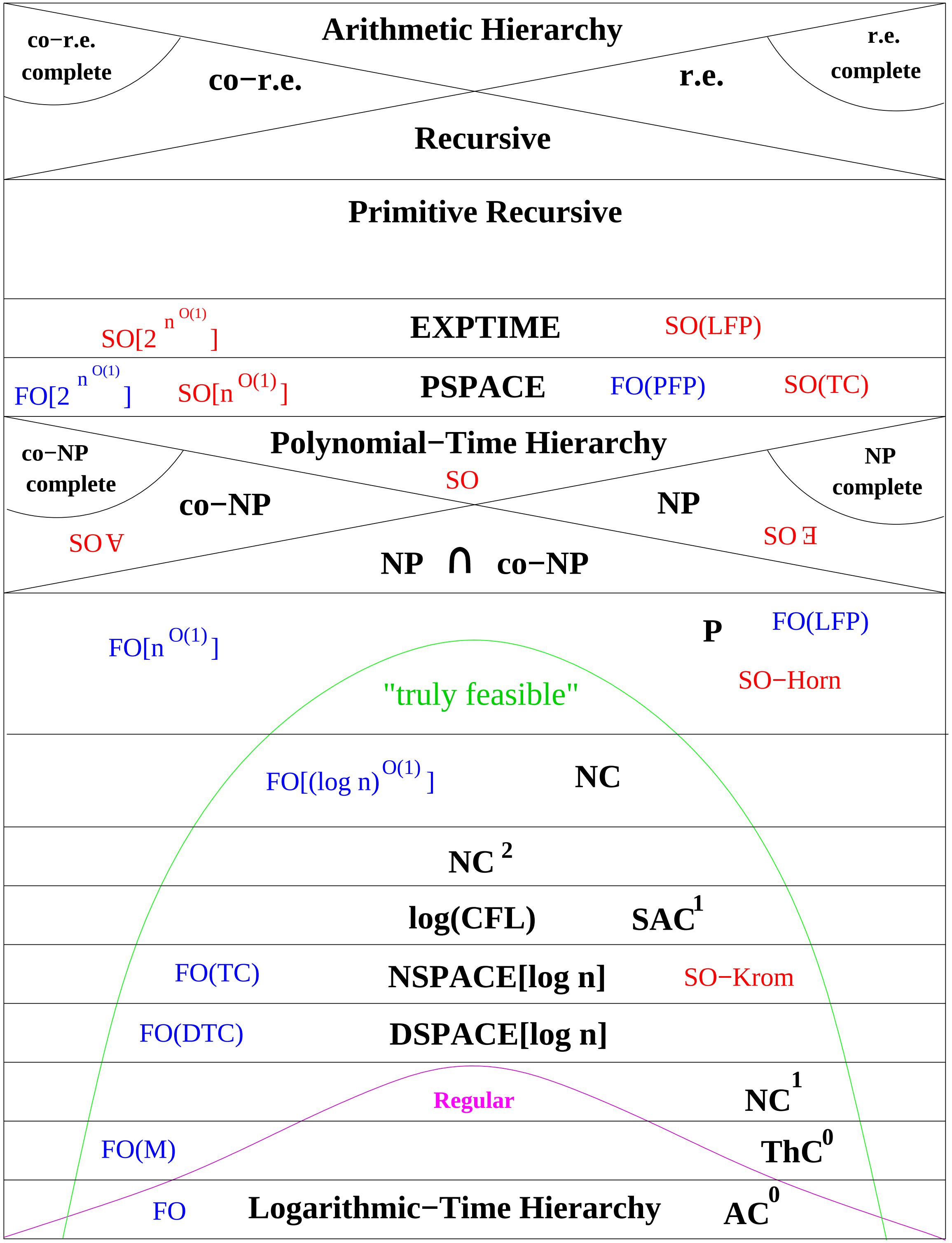
FO(M)

ThC^0

FO

Logarithmic−Time Hierarchy

AC^0



Arithmetic Hierarchy

co−r.e. complete

co−r.e.

r.e.

r.e. complete

Recursive

Primitive Recursive

$SO[2^{n^{O(1)}}]$ **EXPTIME** SO(LFP)

$FO[2^{n^{O(1)}}]$ $SO[n^{O(1)}]$ **PSPACE** FO(PFP) SO(TC)

Polynomial−Time Hierarchy

co−NP complete

SO

NP complete

co−NP

NP

$SO\forall$

$SO\exists$

NP ∩ co−NP

$FO[n^{O(1)}]$ **P** FO(LFP)

"truly feasible"

SO−Horn

$FO[(\log n)^{O(1)}]$ **NC**

NC^2

log(CFL) SAC^1

FO(TC) **NSPACE[log n]** SO−Krom

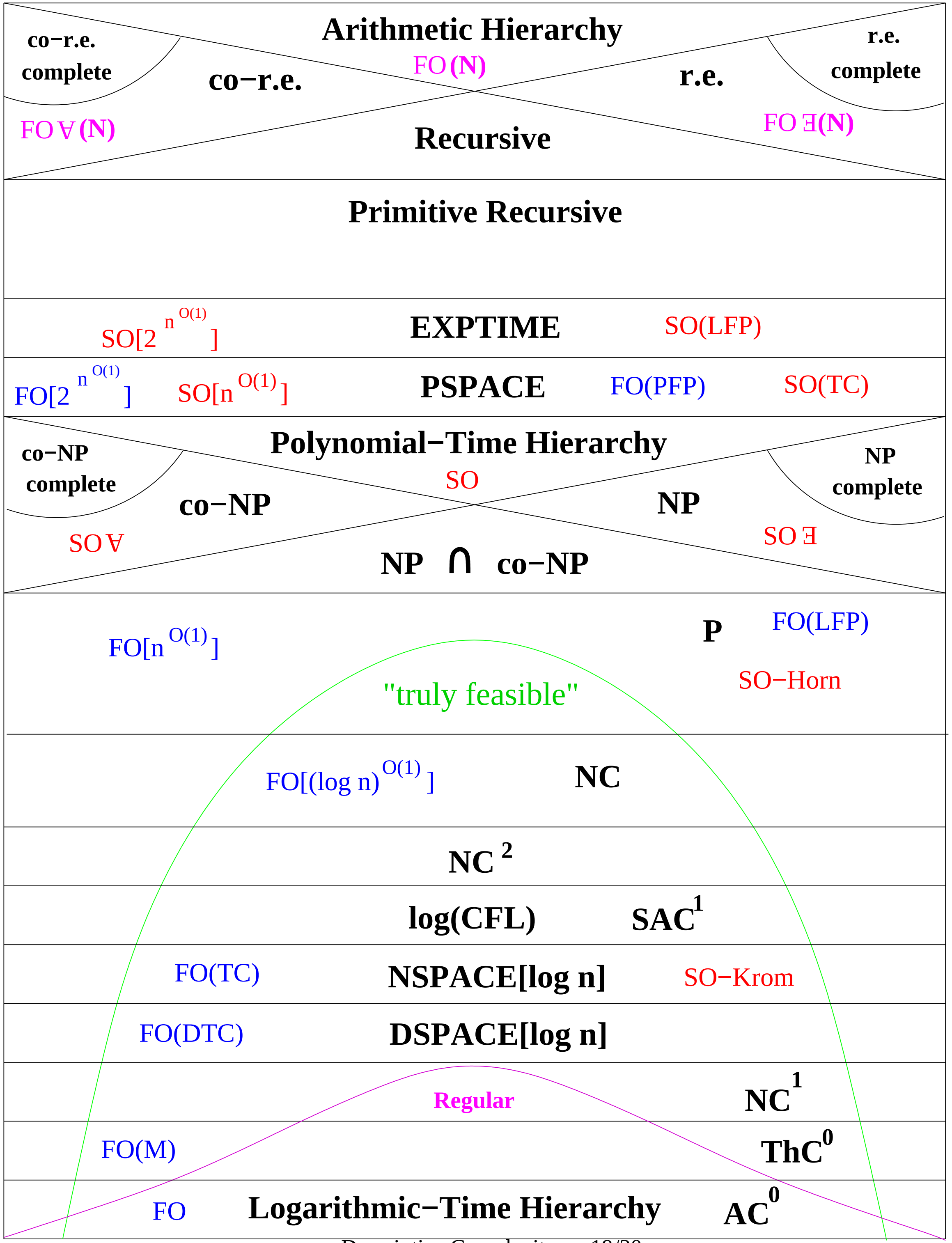
FO(DTC) **DSPACE[log n]**

Regular

NC^1

FO(M) ThC^0

FO **Logarithmic−Time Hierarchy** AC^0



co−r.e. complete

Arithmetic Hierarchy

FO(**N**)

r.e. complete

co−r.e.

r.e.

FO ∀(**N**)

Recursive

FO ∃(**N**)

Primitive Recursive

SO$[2^{n^{O(1)}}]$

EXPTIME

SO(LFP)

FO$[2^{n^{O(1)}}]$

SO$[n^{O(1)}]$

PSPACE

FO(PFP)

SO(TC)

co−NP complete

Polynomial−Time Hierarchy

SO

NP complete

co−NP

NP

SO ∀

SO ∃

NP ∩ co−NP

FO$[n^{O(1)}]$

P

FO(LFP)

"truly feasible"

SO−Horn

FO$[(\log n)^{O(1)}]$

NC

NC^2

log(CFL)

SAC^1

FO(TC)

NSPACE[log n]

SO−Krom

FO(DTC)

DSPACE[log n]

Regular

NC^1

FO(M)

ThC^0

FO

Logarithmic−Time Hierarchy

AC^0

We Need an Ordering on the Universe

$$G = (V, E); \quad V = \{0, 1, \ldots, n-1\}; \quad 0 < 1 < \cdots < n-1$$

An unordered graph makes sense mathematically, but you can't store such an object in a computer as far as I know.

If you remove the ordering then the first-order descriptive characterizations fail:

$\mathbf{EVEN}$ requires $\Omega(n)$ variables without ordering.

Thus, $\mathbf{EVEN} \notin \mathbf{FO}(\mathrm{wo}{\leq})[2^{n^{O(1)}}]; \quad (\mathbf{FO}[2^{n^{O(1)}}] = \mathbf{PSPACE})$

Fagin (SO∃ $=$ $\mathbf{NP}$) didn't run into this problem because in SO∃ you can guess an ordering (unless we are dealing with SO∃(monadic)).

Ehrenfeucht-Fraïssé Games

Delilah: hide any differences between the two structures

Two-person combinatorial game for characterizing what is expressible in a given quantifier depth.

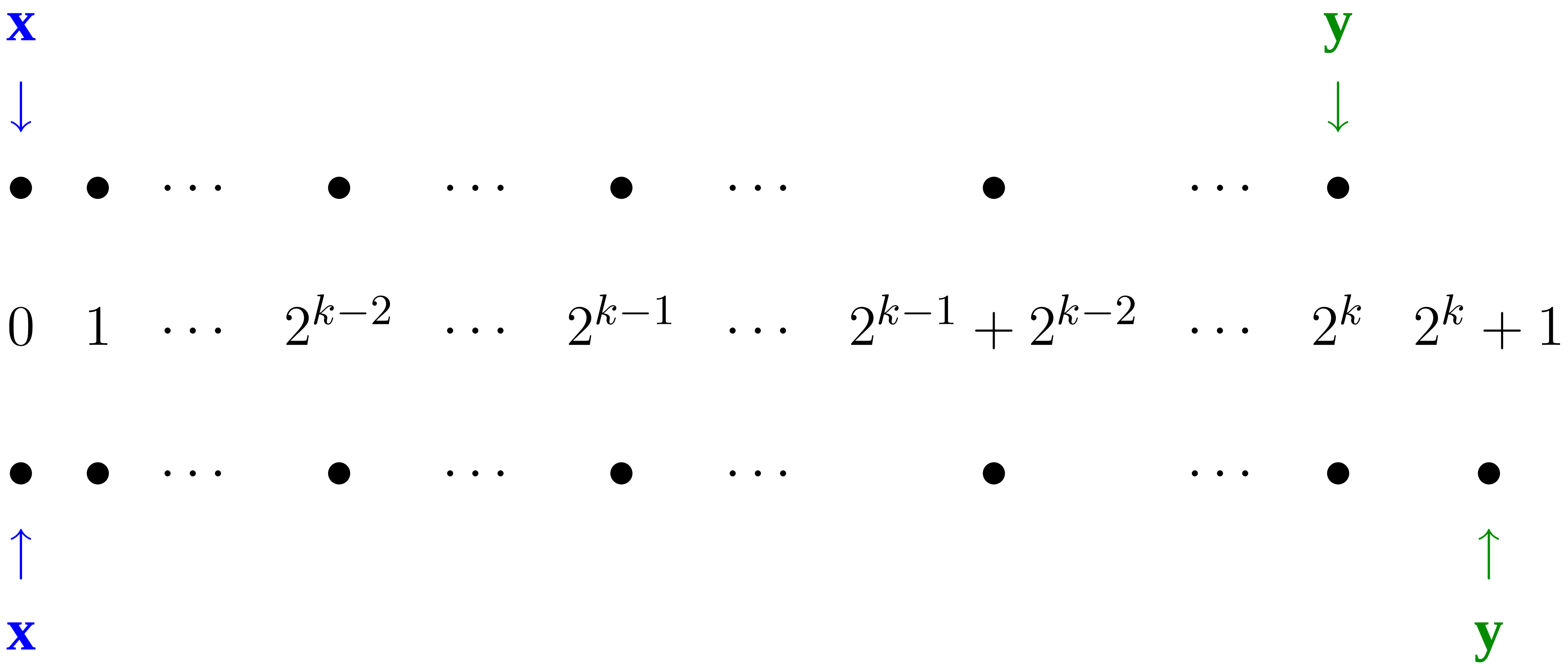
Fundamental Thm: **D** has a winning strategy on the m-move, k-pebble game on $\mathcal{A}, \mathcal{B}$ iff $\mathcal{A}$ and $\mathcal{B}$ agree on all formulas using k variables and quantifier depth m.

$$\mathcal{A} \sim_m^k \mathcal{B} \quad \Leftrightarrow \quad \mathcal{A} \equiv_m^k \mathcal{B}$$

Samson: show a difference

$$\mathrm{dist}(x,y) \leq 2^k \qquad \in \mathbf{FO}^3_k \qquad \notin \mathbf{FO}_{k-1}$$

move 0



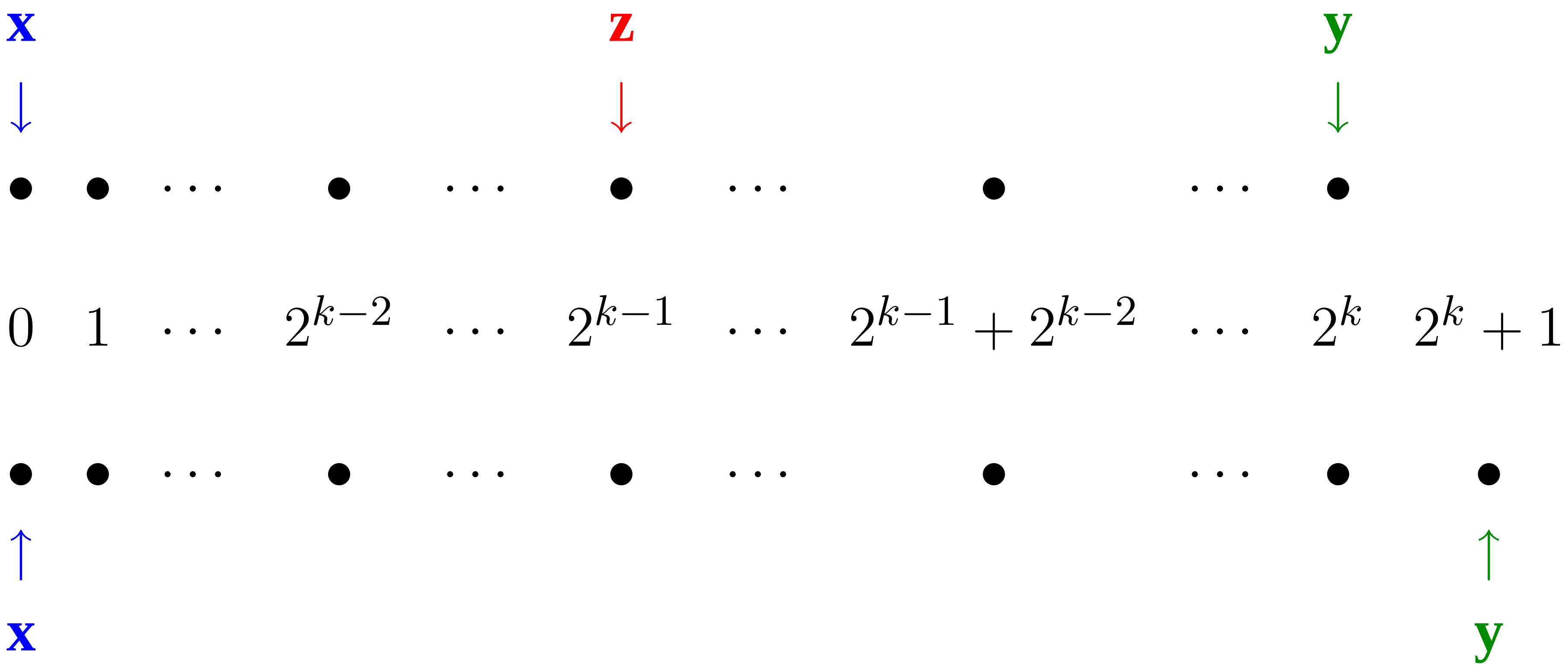
x **y**

$0 \quad 1 \quad \cdots \quad 2^{k-2} \quad \cdots \quad 2^{k-1} \quad \cdots \quad 2^{k-1}+2^{k-2} \quad \cdots \quad 2^k \quad 2^k+1$

x **y**

$$\mathrm{dist}(x,y) \le 2^k \qquad \in \mathbf{FO}^3_k \qquad \not\in \mathbf{FO}_{k-1}$$

move 1 $\quad \exists z(\mathrm{dist}(x,z) \le 2^{k-1} \;\wedge\; \mathrm{dist}(z,y) \le 2^{k-1})$



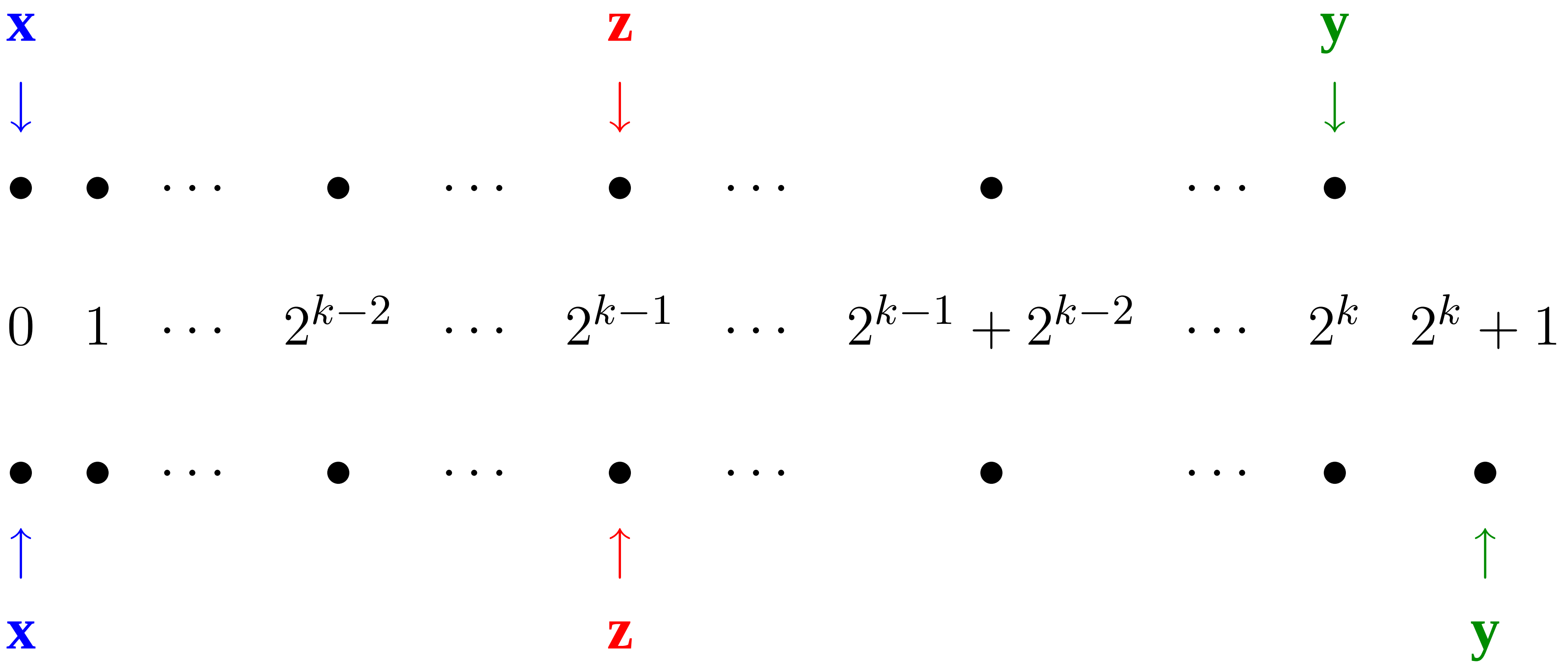
x **z** **y**

$$0 \quad 1 \quad \cdots \quad 2^{k-2} \quad \cdots \quad 2^{k-1} \quad \cdots \quad 2^{k-1}+2^{k-2} \quad \cdots \quad 2^k \quad 2^k+1$$

x **y**

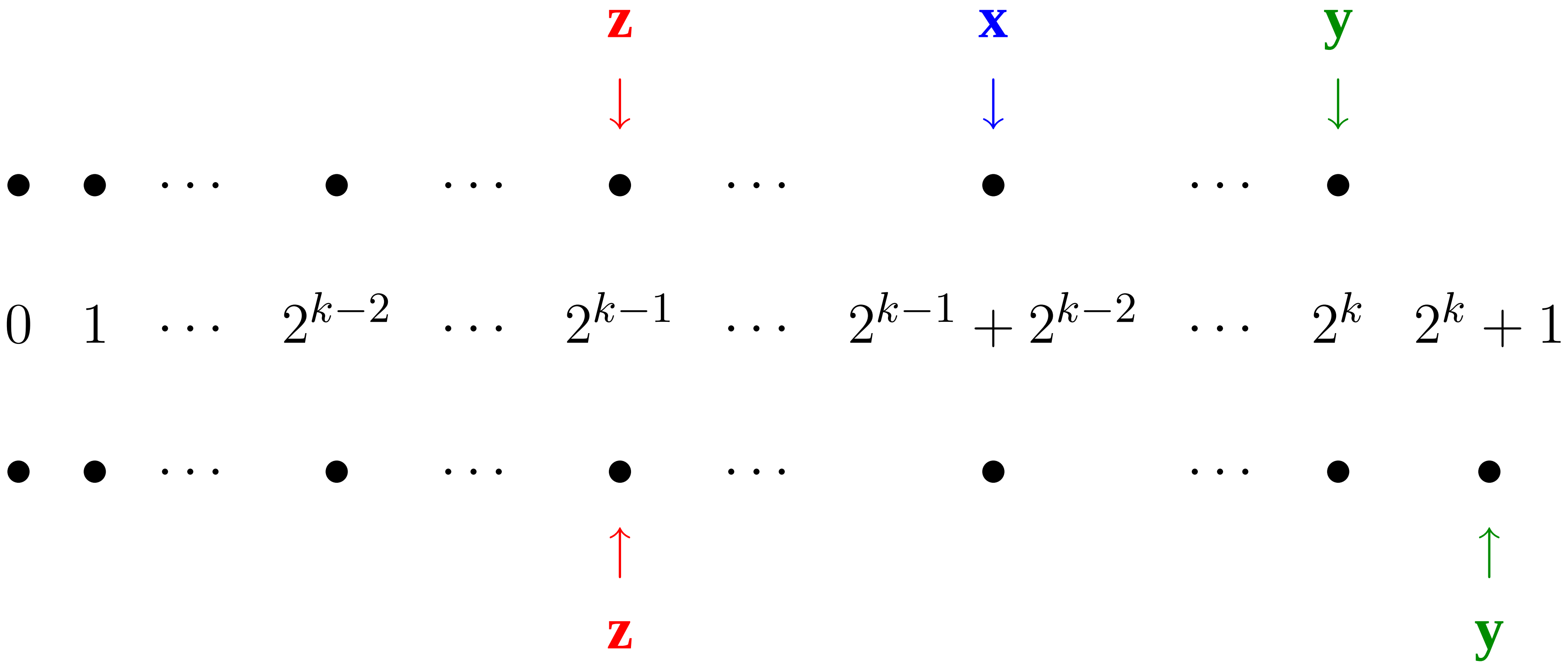
$$\mathrm{dist}(x,y) \le 2^k \quad \in \mathbf{FO}^3_k \quad \notin \mathbf{FO}_{k-1}$$

move 1 $\exists z(\mathrm{dist}(x,z) \le 2^{k-1} \;\wedge\; \mathrm{dist}(z,y) \le 2^{k-1})$



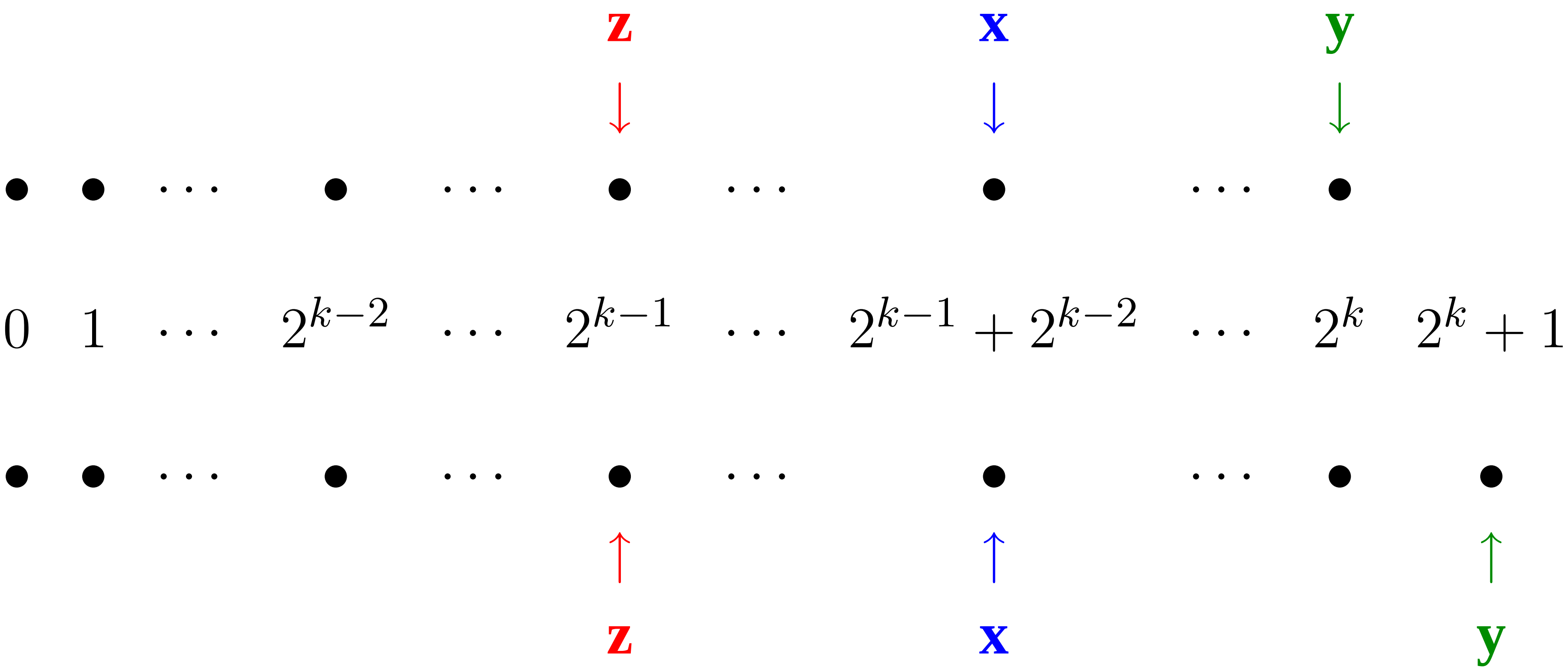
$$\mathrm{dist}(x,y) \le 2^k \qquad \in \mathbf{FO}^3_k \qquad \notin \mathbf{FO}_{k-1}$$

move 2 $\quad \exists x(\mathrm{dist}(z,x) \le 2^{k-2} \;\wedge\; \mathrm{dist}(x,y) \le 2^{k-2})$



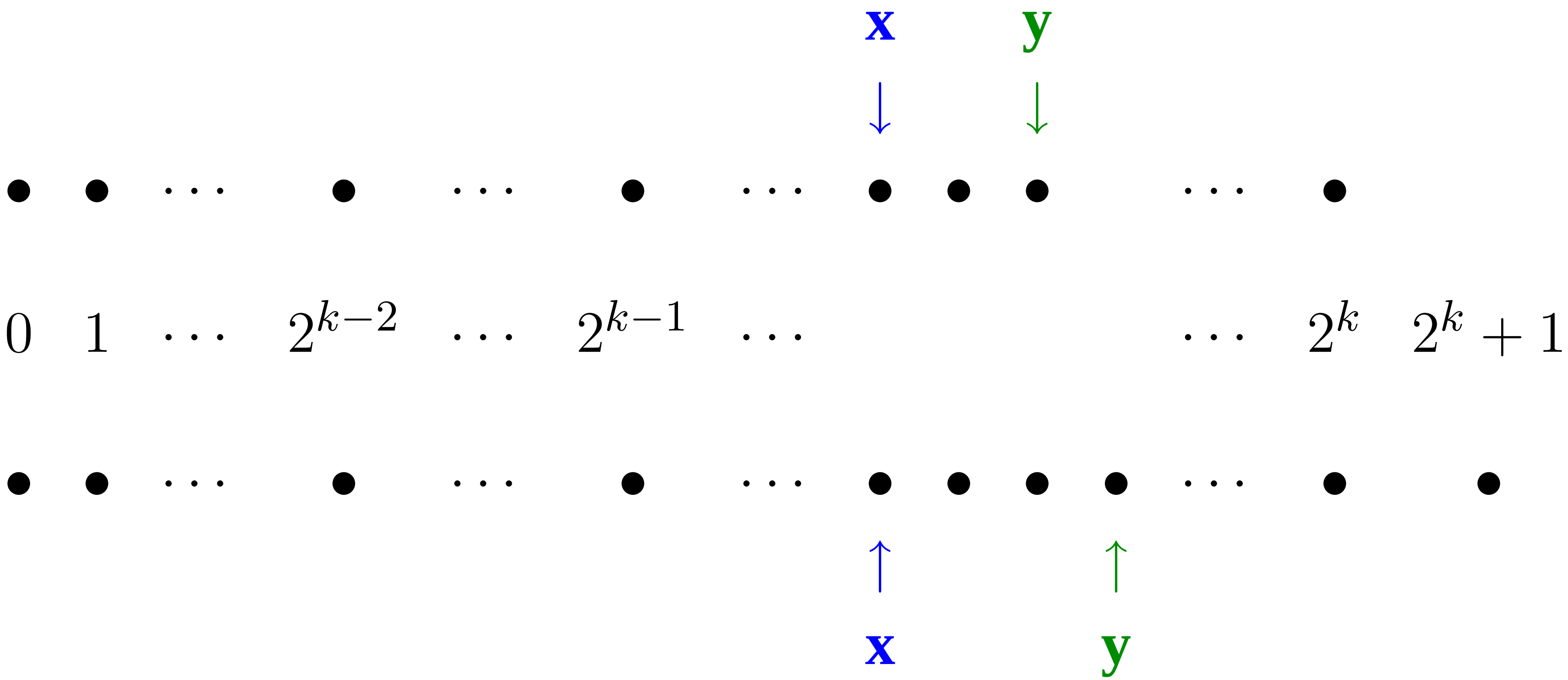
$$\mathrm{dist}(x,y) \le 2^k \quad \in \mathbf{FO}^3_k \quad \notin \mathbf{FO}_{k-1}$$

move 2 $\exists x(\mathrm{dist}(z,x) \le 2^{k-2} \ \wedge \ \mathrm{dist}(x,y) \le 2^{k-2})$



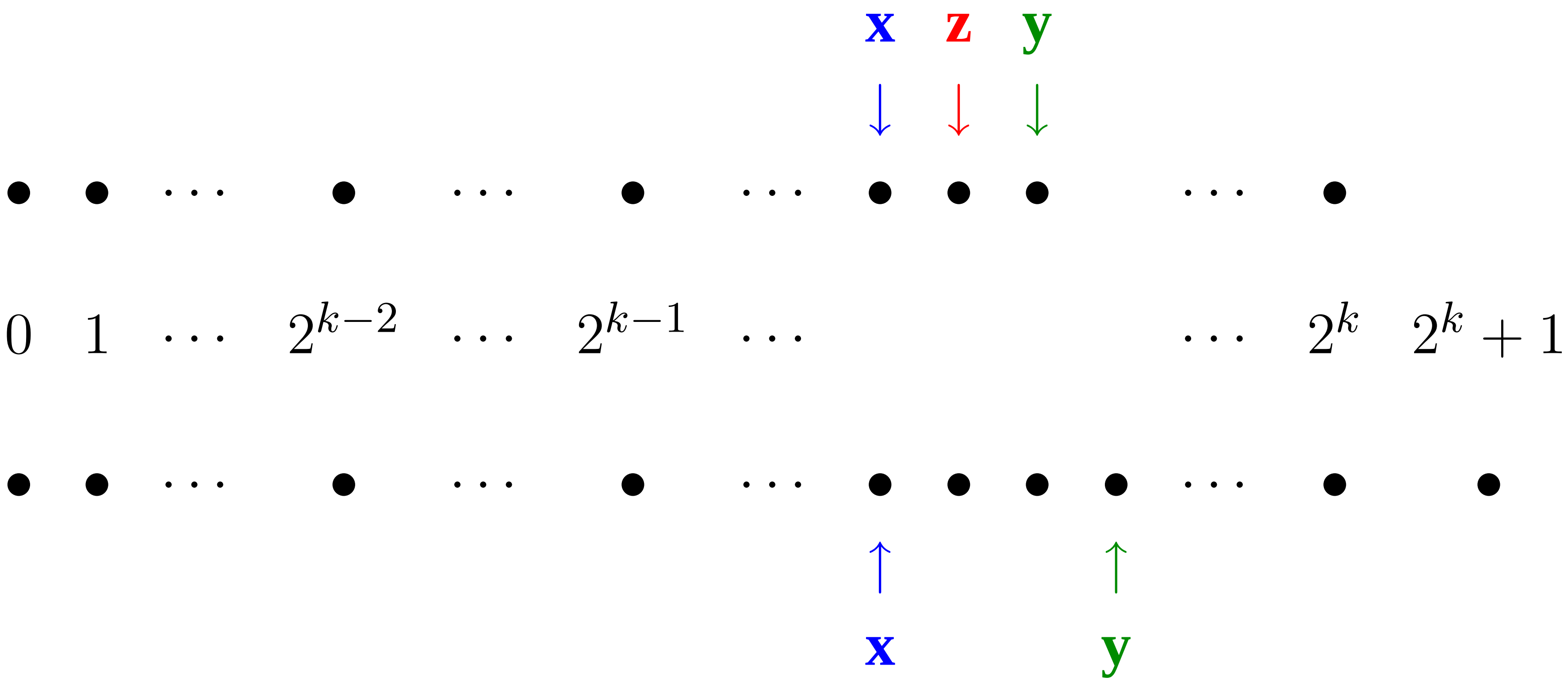
$$\mathrm{dist}(x,y) \leq 2^k \qquad \in \mathbf{FO}^3_k \qquad \notin \mathbf{FO}_{k-1}$$

move k-1: Delilah wins $\quad \mathrm{dist}(x,y) \leq 2^1$



$$\mathrm{dist}(x,y) \leq 2^k \qquad \in \mathbf{FO}^3_k \qquad \notin \mathbf{FO}_{k-1}$$

move k: Samson wins $\qquad E(x,z) \wedge E(z,y)$



These games have been used to prove many beautiful lower bounds without the ordering, and thus separate most descriptive classes without ordering. Among others:

- McColm and Grädel: $\mathbf{DTC}, \mathbf{TC}, \mathbf{LFP}$
- Grohe: arity hierarchies for $\mathbf{TC}$ and $\mathbf{LFP}$
- Etessami: lower bounds with counting and one-way local orderings
- Libkin: lower bounds with counting via locality arguments
- Grohe and Schwentick: locality of order-independent queries

One Historical Thread

- Fagin: $\mathrm{REACH} \notin$ SO∃(monadic) and thus since $\overline{\mathrm{REACH}} \in$ SO∃(monadic), SO∃(monadic) is not closed under complementation.
- de Rougemont: remains true with successor
- Schwentick: remains true with ordering!
- Doesn't give us a complexity lower bound, cf., Lynch: $\mathbf{NTIME}[n^k] \subseteq$ SO∃$(+)(\text{arity } k)$

Ehrenfeucht-Fraïssé games: lower bounds on depth.

With ordering, depth $2 + \log n$ suffices to express any graph property for graphs on n vertices.   Let $G = (V, E)$

$\#_i(x) \;\equiv\;$ "exists a path of length i from 0 to x" $\in \mathrm{depth}(\log n)$

$$\varphi_G \quad \equiv \quad \bigwedge_{\langle i,j\rangle \in E} \exists x, y(\#_i(x) \wedge \#_j(y) \wedge E(x,y)) \quad \wedge$$

$$\bigwedge_{\langle i,j\rangle \notin E} \exists x, y(\#_i(x) \wedge \#_j(y) \wedge \neg E(x,y))$$

For S an arbitrary set of graphs on n vertices:

$$\varphi_S \;\equiv\; \bigvee_{G \in S} \varphi_G; \qquad \mathrm{depth}(\varphi_S) = 2 + \log n$$

Newish Size Lower Bound Game

Adler & I defined these games and proved a pair of optimal succinctness result for temporal logics.

Karchmer and Wigderson used similar communication complexity games to prove lower bound on monotone circuits for REACH.

Idea: unlike Ehrenfeucht-Fraïssé games, we play on a pair of sets of structures, A, B. We determine the size of the smallest sentence true of all of A and none of B.

More recent lower bounds by Grohe and Schweikardt

Newish Size Lower Bound Game

Adler & I defined these games and proved a pair of optimal succinctness result for temporal logics.

Karchmer and Wigderson used similar communication complexity games to prove lower bound on monotone circuits for REACH.

Idea: unlike Ehrenfeucht-Fraïssé games, we play on a pair of sets of structures, A, B. We determine the size of the smallest sentence true of all of A and none of B.

More recent lower bounds by Grohe and Schweikardt

Not surprisingly, these games are much harder to play than Ehrenfeucht-Fraïssé games . . .

Size versus Number of Variables on Line Graphs

$$\begin{aligned} \text{dist}(x,y) \le 1 &\equiv x = y \ \vee \ E(x,y) \\ \text{dist}(x,y) \le 2k &\equiv \exists z(\text{dist}(x,z) \le k \ \wedge \ \text{dist}(z,y) \le k) \end{aligned}$$

$$\text{dist}(x,y) \le n \in \mathbf{FO}^3_{\log n}; \quad \mathbf{SIZE}(n)$$

Size versus Number of Variables on Line Graphs

$$\begin{aligned}\text{dist}(x,y) \le 1 &\equiv x = y \ \lor \ E(x,y)\\ \text{dist}(x,y) \le 2k &\equiv \exists z(\text{dist}(x,z) \le k \ \land \ \text{dist}(z,y) \le k)\end{aligned}$$

$$\text{dist}(x,y) \le n \quad \in \quad \mathbf{FO}^3_{\log n}; \quad \mathbf{SIZE}(n)$$

$$\begin{aligned}\text{dist}_u(x,y) \le 1 &\equiv x = y \ \lor \ E(x,y) \ \lor \ E(y,x)\\ \text{dist}_u(x,y) \le 2k &\equiv \exists z \forall w((w = x \lor w = y) \to \text{dist}_u(z,w) \le k)\end{aligned}$$

$$\text{dist}_u(x,y) \le n \quad \in \quad \mathbf{FO}^4_{\log n}; \quad \mathbf{SIZE}(O(\log n))$$

Size versus Number of Variables on Line Graphs

$$\begin{aligned}\mathrm{dist}(x,y) \le 1 &\equiv x = y \lor E(x,y)\\ \mathrm{dist}(x,y) \le 2k &\equiv \exists z(\mathrm{dist}(x,z) \le k \land \mathrm{dist}(z,y) \le k)\end{aligned}$$

$$\mathrm{dist}(x,y) \le n \quad \in \quad \mathbf{FO}^3_{\log n}; \quad \mathbf{SIZE}(n)$$

$$\begin{aligned}\mathrm{dist}_u(x,y) \le 1 &\equiv x = y \lor E(x,y) \lor E(y,x)\\ \mathrm{dist}_u(x,y) \le 2k &\equiv \exists z \forall w((w = x \lor w = y) \rightarrow \mathrm{dist}_u(z,w) \le k)\end{aligned}$$

$$\mathrm{dist}_u(x,y) \le n \quad \in \quad \mathbf{FO}^4_{\log n}; \quad \mathbf{SIZE}(O(\log n))$$

Grohe-Schweikardt:

- $\mathbf{FO}^2$ and $\mathbf{FO}^3$ are polynomially **SIZE**-related
- Exponential Size gap between $\mathbf{FO}^3$ and $\mathbf{FO}^4$
- Gap between $\mathbf{FO}^k$ and $\mathbf{FO}^{k+1}$ is open for $k > 3$

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- $\mathbf{FO}$ on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- **FO** on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

Rankers

$b \quad a \quad c \quad d \quad b \quad b \quad a \quad c \quad d$

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- **FO** on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

Rankers

$b \quad a \quad c \quad d \quad b \quad b \quad a \quad c \quad d$ $\qquad\qquad \triangleright a$

$\quad \uparrow$ $\qquad\qquad$ 1 ranker

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- $\mathbf{FO}$ on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

Rankers

$b \quad a \quad c \quad d \quad b \quad b \quad a \quad c \quad d$ $\qquad\qquad$ $\triangleright a \ \triangleright b$

$\uparrow$ (under the fifth letter) $\qquad$ 2 ranker

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- **FO** on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

Rankers

$$\begin{array}{ccccccccc} b & a & c & d & b & b & a & c & d \\ & & \uparrow & & & & & & \end{array}$$

$$\triangleright a \ \ \triangleright b \ \triangleleft c$$

3 ranker

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- **FO** on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

Rankers

$b \quad a \quad c \quad d \quad b \quad b \quad a \quad c \quad d$

$\triangleright a \ \triangleright b \ \triangleleft c \ \triangleleft d$

4 ranker

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- **FO** on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

Rankers

$$\begin{array}{ccccccccc} b & a & c & d & b & b & a & c & d \\ & & \uparrow & & & & & & \end{array}$$

$\triangleright a \;\; \triangleright b \;\; \triangleleft c$

3 ranker

Thm: $\mathrm{FO}^2_m \equiv$ which m-rankers exist, and, relative order with smaller rankers.

FO^2 on Words: [Philipp Weis & I]

- **FO** on words reduces to **FO**^{3} on words
- **FO**^{2} on words was well studied, except alternation hierarchy was open

Rankers

$b \quad a \quad c \quad d \quad b \quad b \quad a \quad c \quad d$ $\qquad\qquad \triangleright a \ \triangleright b \ \triangleleft c$

$\qquad\quad \uparrow$ $\qquad\qquad$ 2, 3 ranker

Thm: FO^2_m $\equiv$ which m-rankers exist, and, relative order with smaller rankers.

Thm: FO$^2_{k,m}$ $\equiv$ which k, m-rankers exist, and, relative order with smaller rankers.

$\mathbf{FO}^2$ on Words: [Philipp Weis & I]

- **FO** on words reduces to $\mathbf{FO}^3$ on words
- $\mathbf{FO}^2$ on words was well studied, except alternation hierarchy was open

Rankers

$$b \quad a \quad c \quad d \quad b \quad b \quad a \quad c \quad d \qquad\qquad \triangleright a \;\; \triangleright b \;\; \triangleleft c$$

$$\uparrow$$

2, 3 ranker

Thm: $\mathrm{FO}^2_m \quad \equiv$ which m-rankers exist, and, relative order with smaller rankers.

Thm: $\mathrm{FO}^2_{k,m} \quad \equiv$ which k, m-rankers exist, and, relative order with smaller rankers.

Thm: FO^2 has a strict alternation hierarchy.

Ultimately, want to understand the VAR-SIZE trade-off in many settings.

Currently, Philipp Weis and I would be happy to settle what Grohe and Schweikardt left open, then to move on to words, and other simple graphs, 2 and 3 variables . . .

Ultimately, want to understand the VAR-SIZE trade-off in many settings.

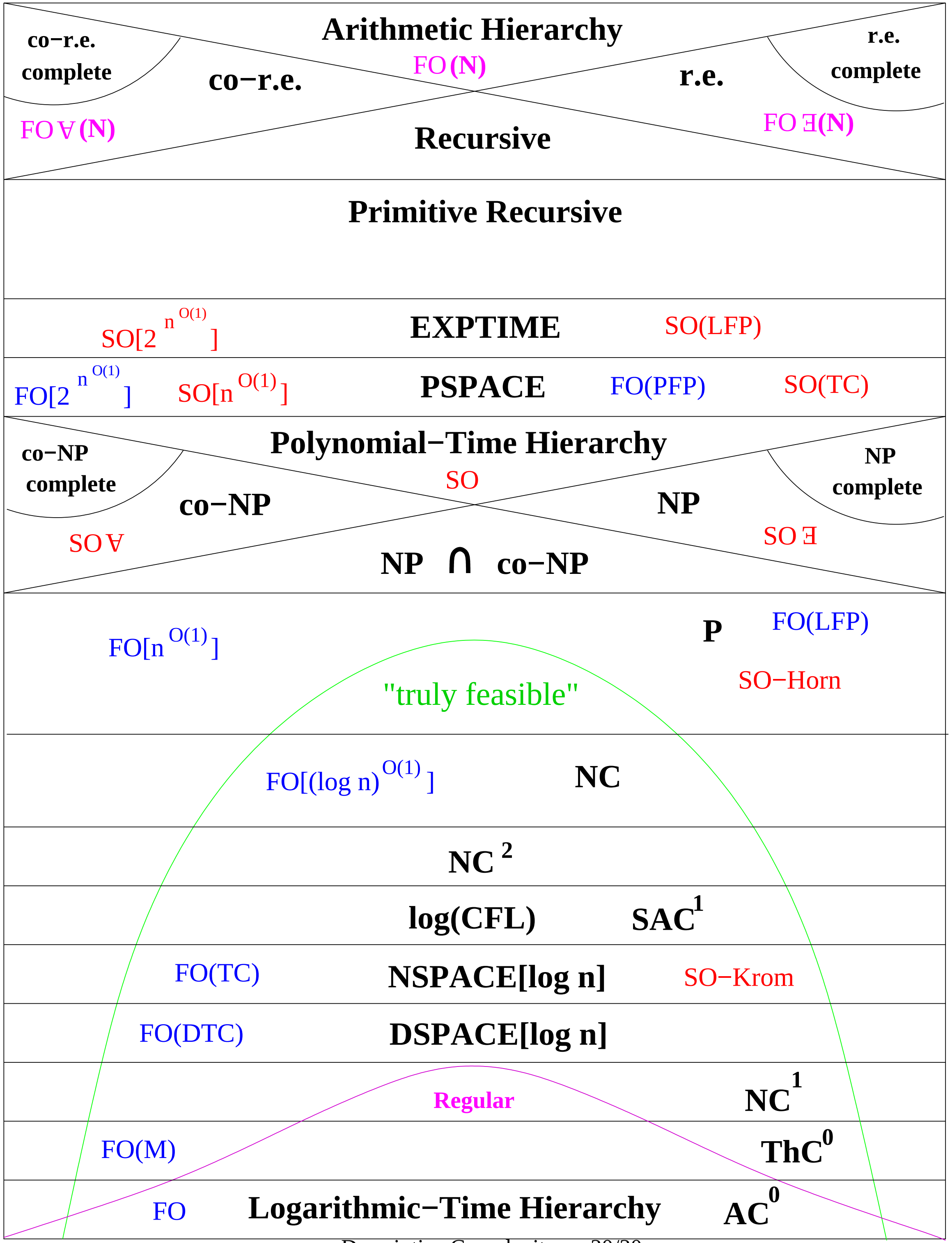
Currently, Philipp Weis and I would be happy to settle what Grohe and Schweikardt left open, then to move on to words, and other simple graphs, 2 and 3 variables . . .

Would one day like to understand the following:

$$\mathbf{PSPACE} \quad = \quad \mathbf{FO}[2^{n^{O(1)}}] \quad = \quad \mathbf{SO}[n^{O(1)}]$$

$$\mathbf{NC}^1 \quad \subseteq \quad \mathbf{FO}[\log n / \log \log n]$$

$$\mathbf{NC}^1 \subseteq \mathbf{L} \subseteq \mathbf{NL} \subseteq \mathbf{sAC}^1 \subseteq \mathbf{AC}^1 = \mathbf{FO}[\log n]$$



Arithmetic Hierarchy

co−r.e. complete

FO $\forall$ (**N**)

co−r.e.

FO (**N**)

r.e.

r.e. complete

FO $\exists$(**N**)

Recursive

Primitive Recursive

SO[$2^{n^{O(1)}}$]

EXPTIME

SO(LFP)

FO[$2^{n^{O(1)}}$]

SO[$n^{O(1)}$]

PSPACE

FO(PFP)

SO(TC)

Polynomial−Time Hierarchy

co−NP complete

co−NP

SO

NP

NP complete

SO $\forall$

SO $\exists$

NP $\cap$ co−NP

FO[$n^{O(1)}$]

P

FO(LFP)

SO−Horn

"truly feasible"

FO[$(\log n)^{O(1)}$]

NC

NC^2

log(CFL)

SAC^1

FO(TC)

NSPACE[log n]

SO−Krom

FO(DTC)

DSPACE[log n]

Regular

NC^1

FO(M)

ThC^0

FO

Logarithmic−Time Hierarchy

AC^0