Separations in Proof Complexity and TFNP

Mika Göös
Siddhartha Jain
Gilbert Maystre
EPFL

William Pires
Robert Robere
Ran Tao
McGill

Alexandros Hollender
Oxford

Separations in Proof Complexity and TFNP

Ran Tao
Robert Robere
McGill

Mika Göös
Gilbert Maystre
Alexandros Hollender
EPFL

William Pires
Columbia

Siddhartha Jain
UT Austin

Separations in Proof Complexity and TFNP

MIAO Seminar, Copenhagen

Mika Göös
Gilbert Maystre
Alexandros Hollender
EPFL

William Pires
Columbia

Ran Tao
Robert Robere
McGill

Siddhartha Jain
UT Austin

Understanding The Title

TFNP := Total Function NP

Polytime $R(x,y)$

TFNP := Total Function NP

Polytime $R(x,y)$

Input x

Output $y : R(x,y) = 1$ & $|y| \leq |x|^{O(1)}$

TFNP := Total Function NP

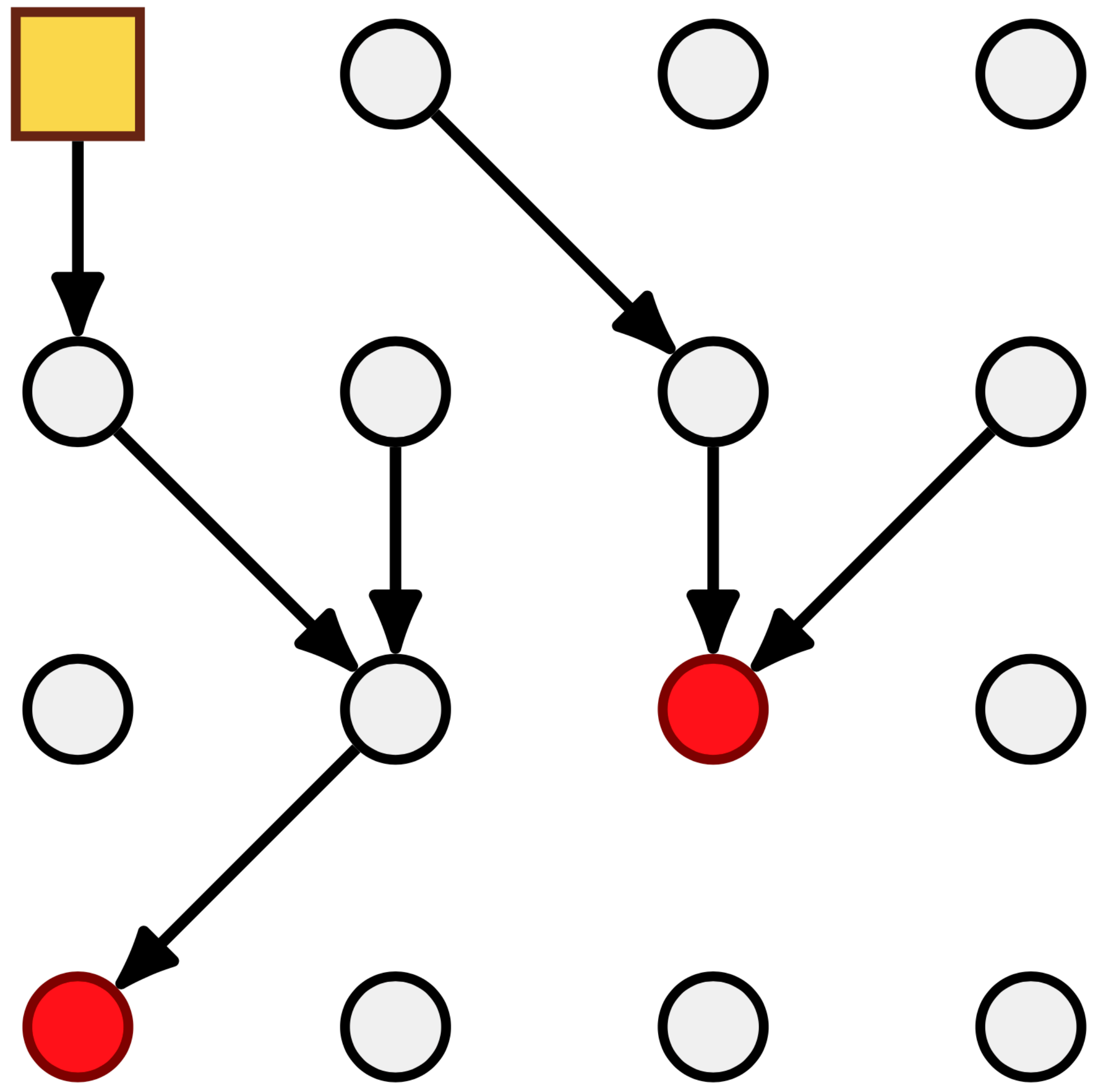
Polytime $R(x,y)$

Input x

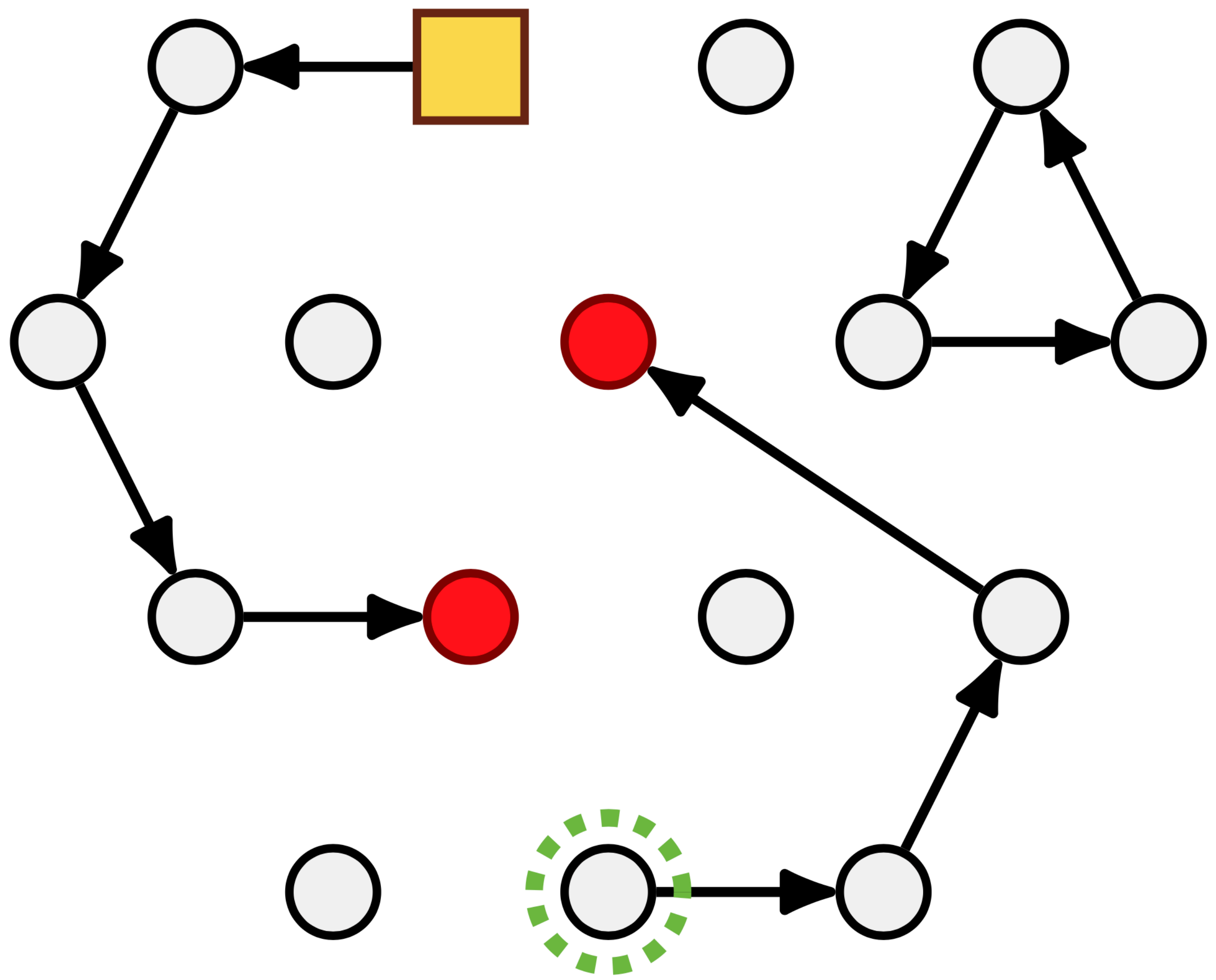
Output $y : R(x,y) = 1$ & $|y| \leq |x|^{O(1)}$

Promise R is total: $\forall x \, \exists y \; R(x,y) = 1$

Two Problems

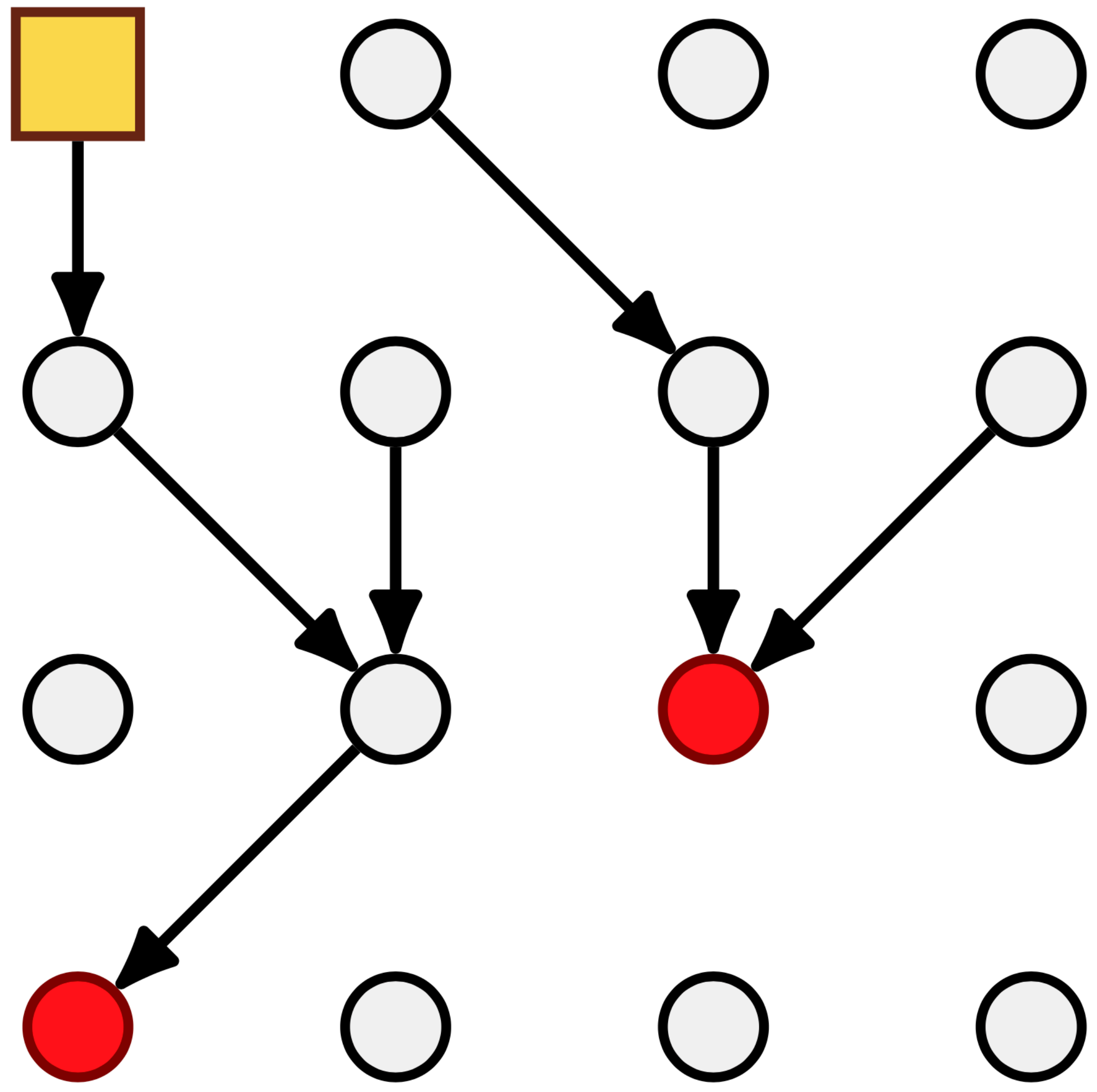


Sink-of-DAG (SoD)

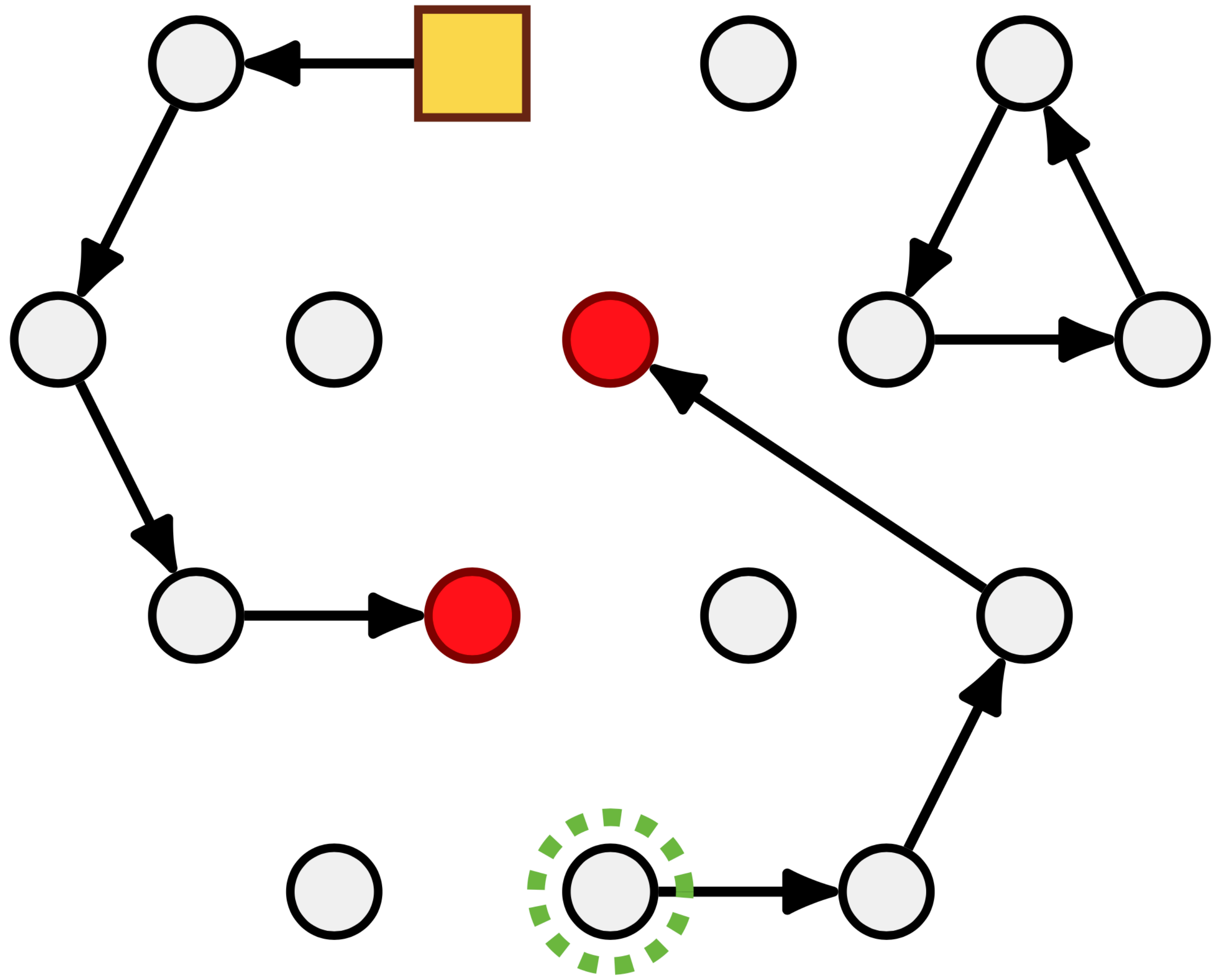


Sink-of-Line (SoL)

Two (& ½) Problems



Sink-of-DAG (SoD)



Sink-of-Line (SoL)
End-of-Line (EoL)

… And Three Classes

$$\mathrm{PLS} = \{ P : P \leq \mathrm{S{\circ}D} \}$$

$$\mathrm{PPADS} = \{ P : P \leq \mathrm{S{\circ}L} \}$$

$$\mathrm{PPAD} = \{ P : P \leq \mathrm{E{\circ}L} \}$$

... And Three Classes

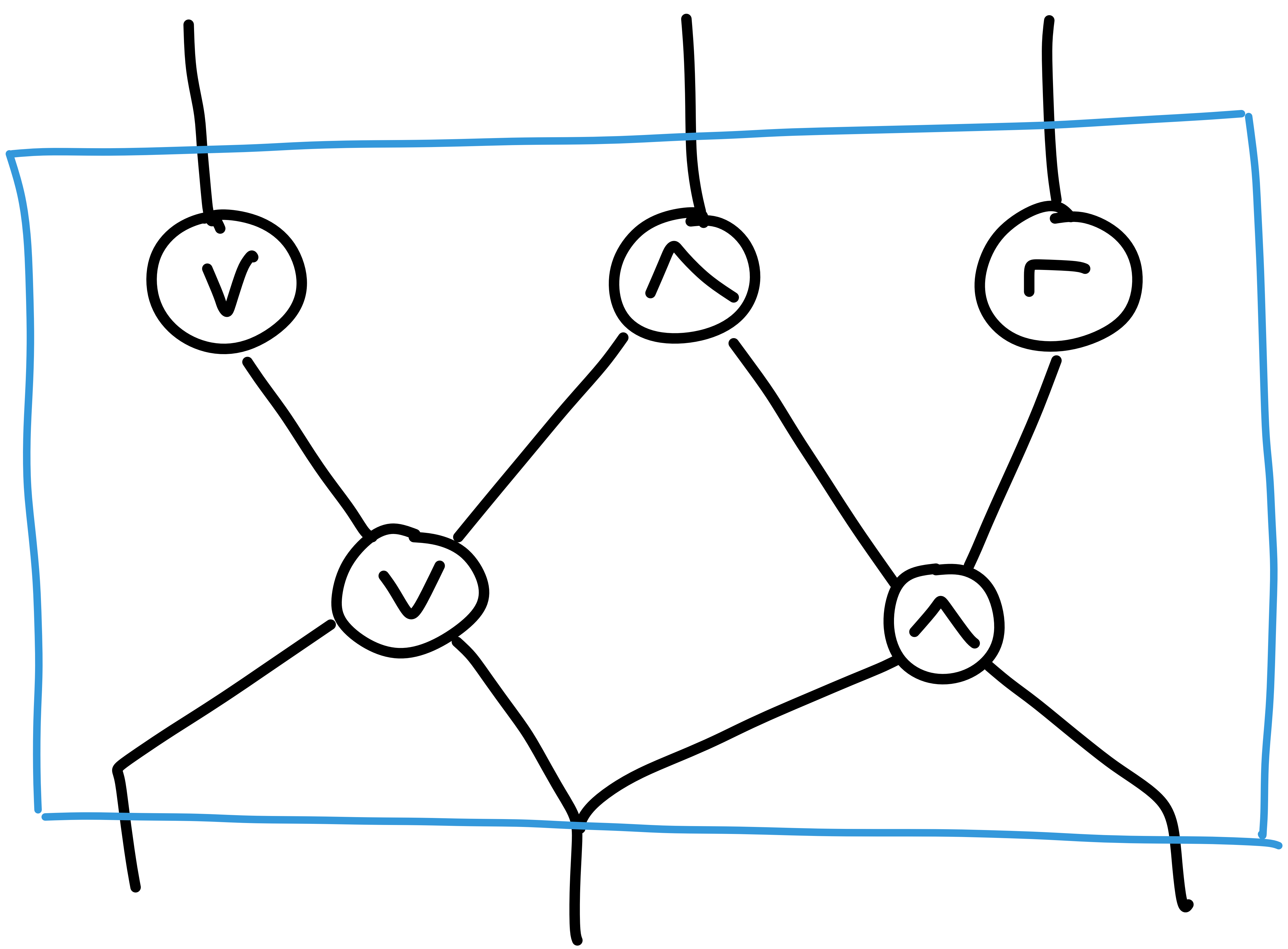
$$\mathrm{PLS} = \{ P : P \leq \mathrm{SoD} \}$$

$$\mathrm{PPADS} = \{ P : P \leq \mathrm{SoL} \}$$

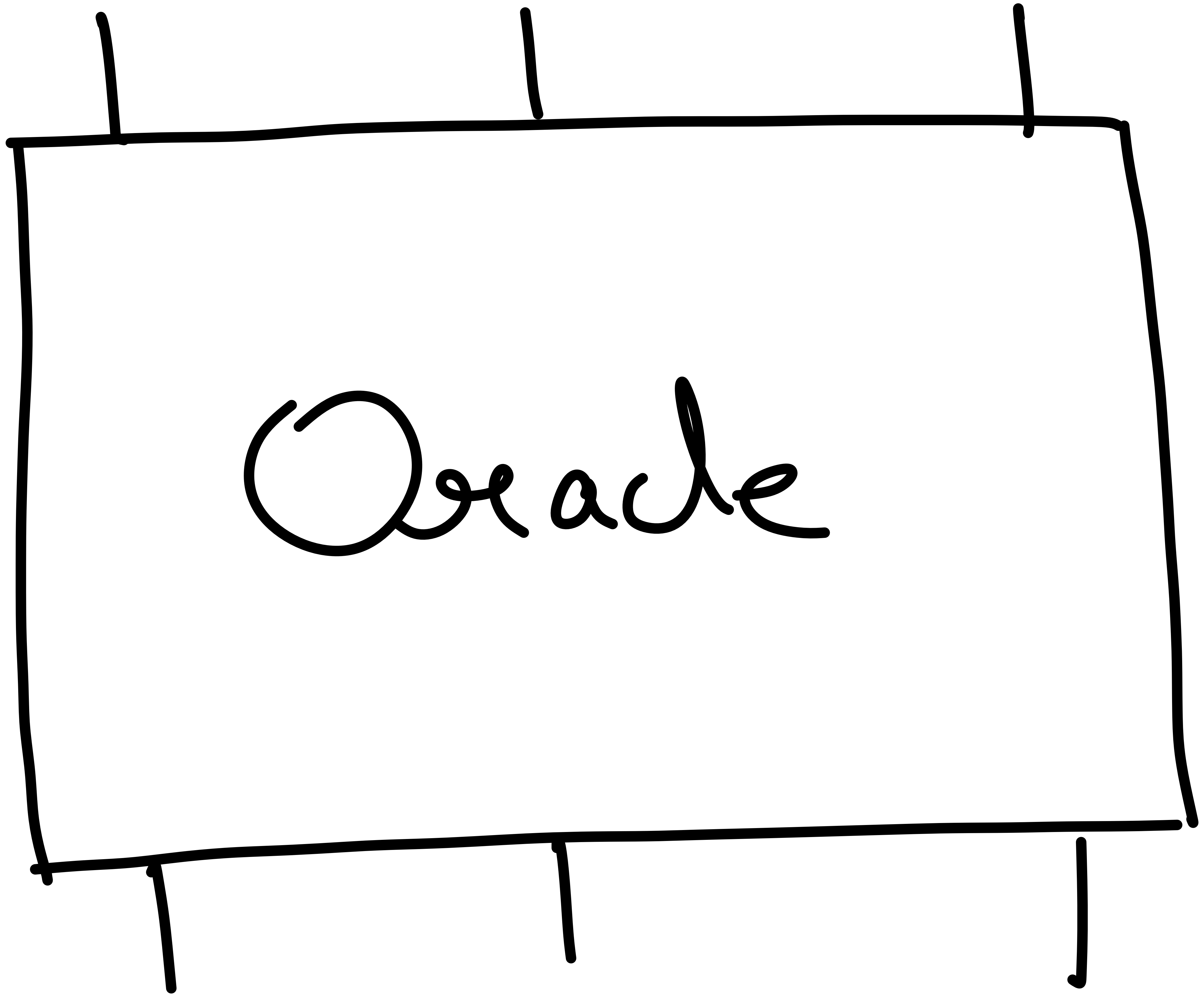
$$\mathrm{PPAD} = \{ P : P \leq \mathrm{EoL} \}$$

$A \leq B$ if $\exists\, R_1, R_2$

$$\begin{tikzcd}
x \arrow[r, "R_1"] \arrow[d, dashed, "\text{Solves } A"'] & x' \arrow[d, "\text{Solve } B"] \\
y & y' \arrow[l, "R_2"]
\end{tikzcd}$$

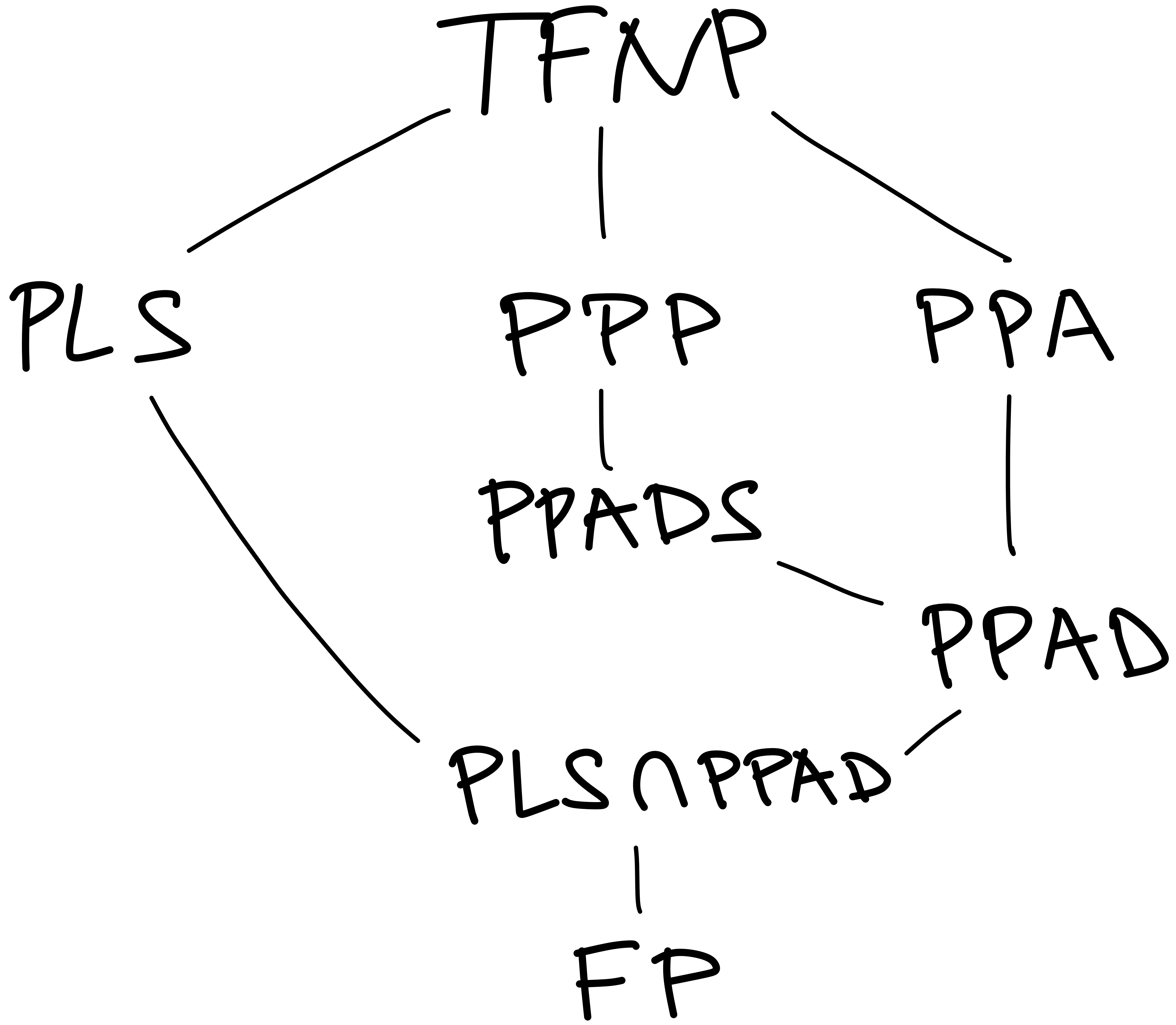


White-box



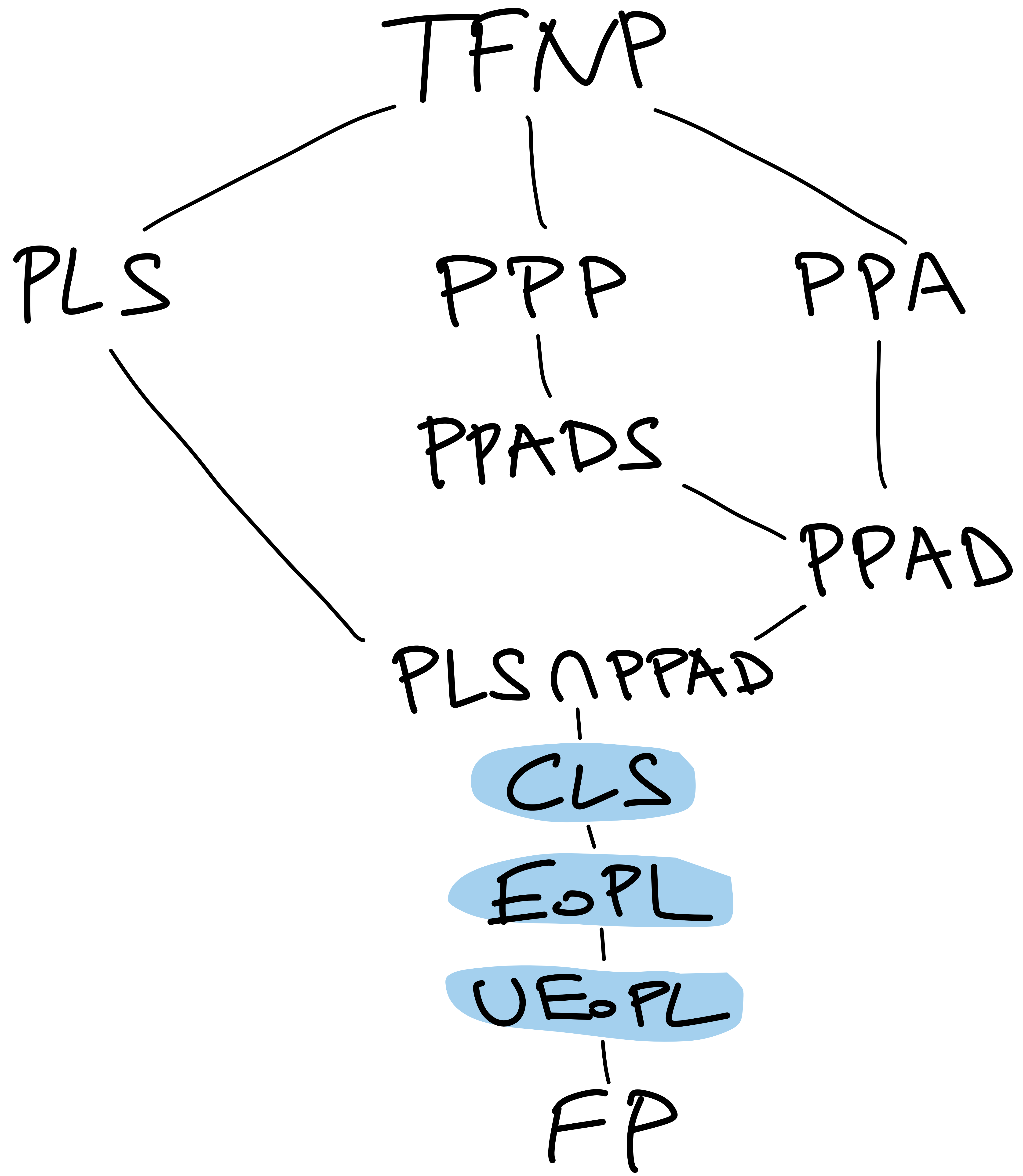
Black-box

Classical hierarchy (90's and 00's)



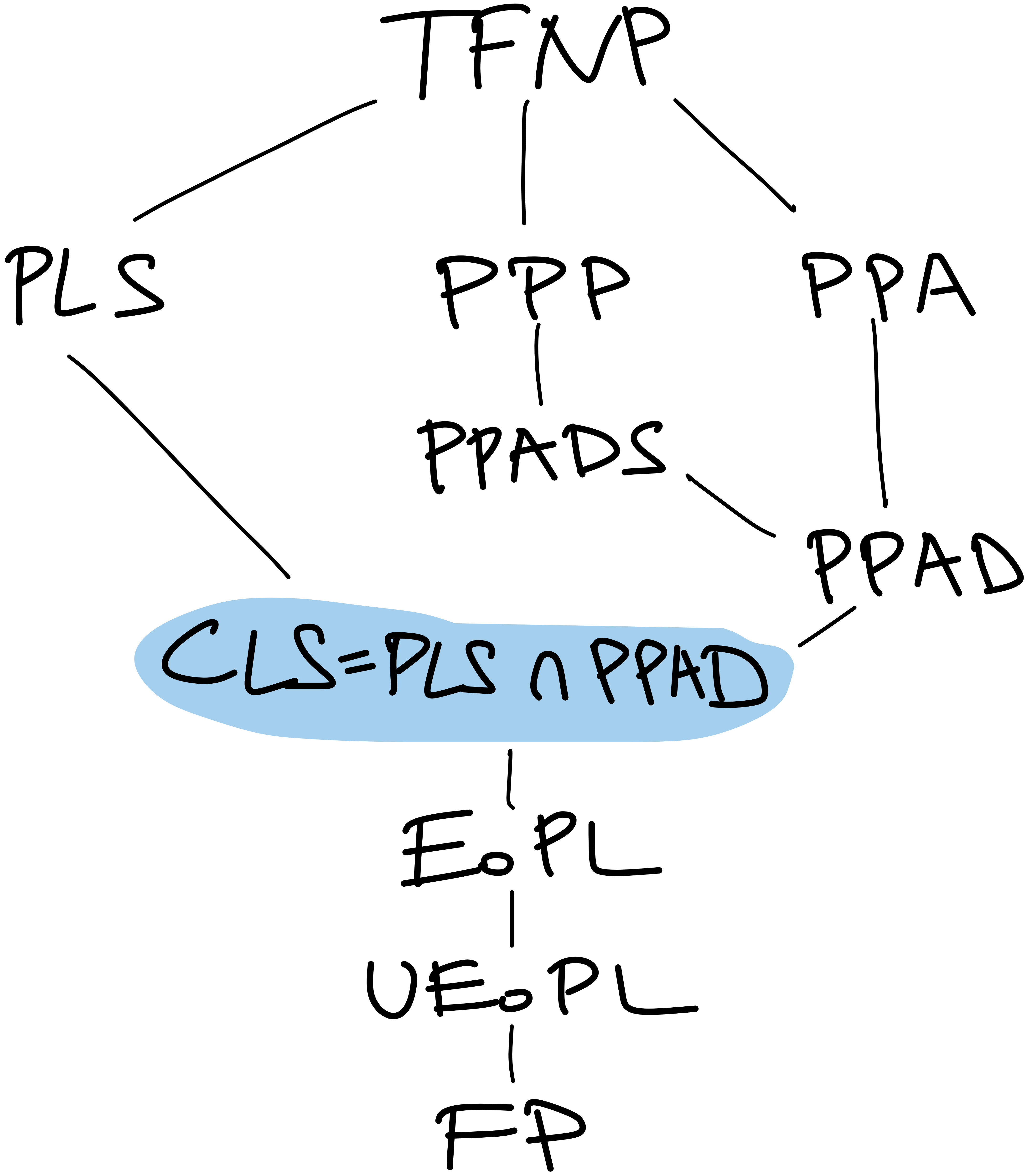
[Pab94]
[JPY88]

New classes (10's)



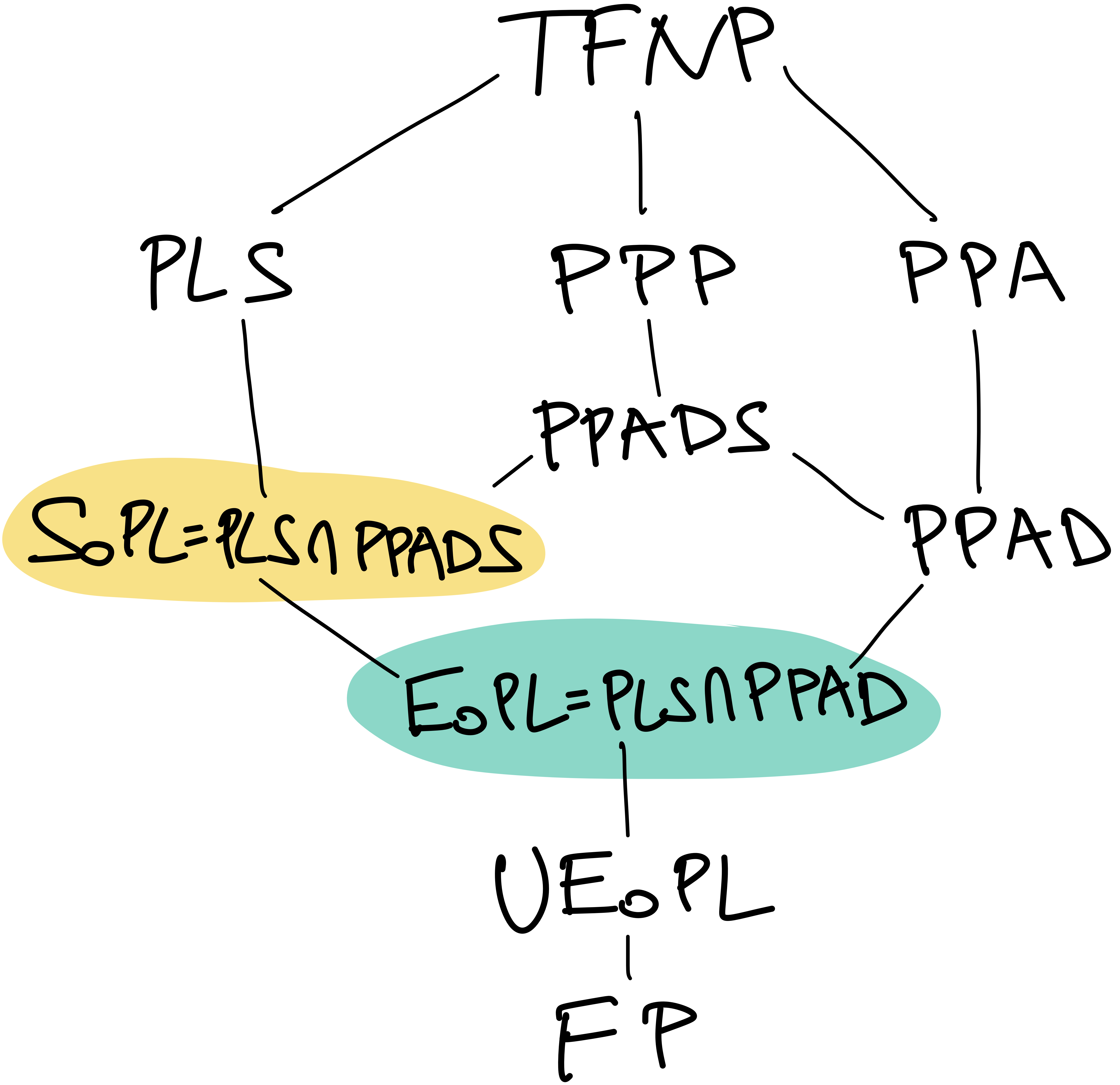
[HY20]
[FGMS20]
[DP11]

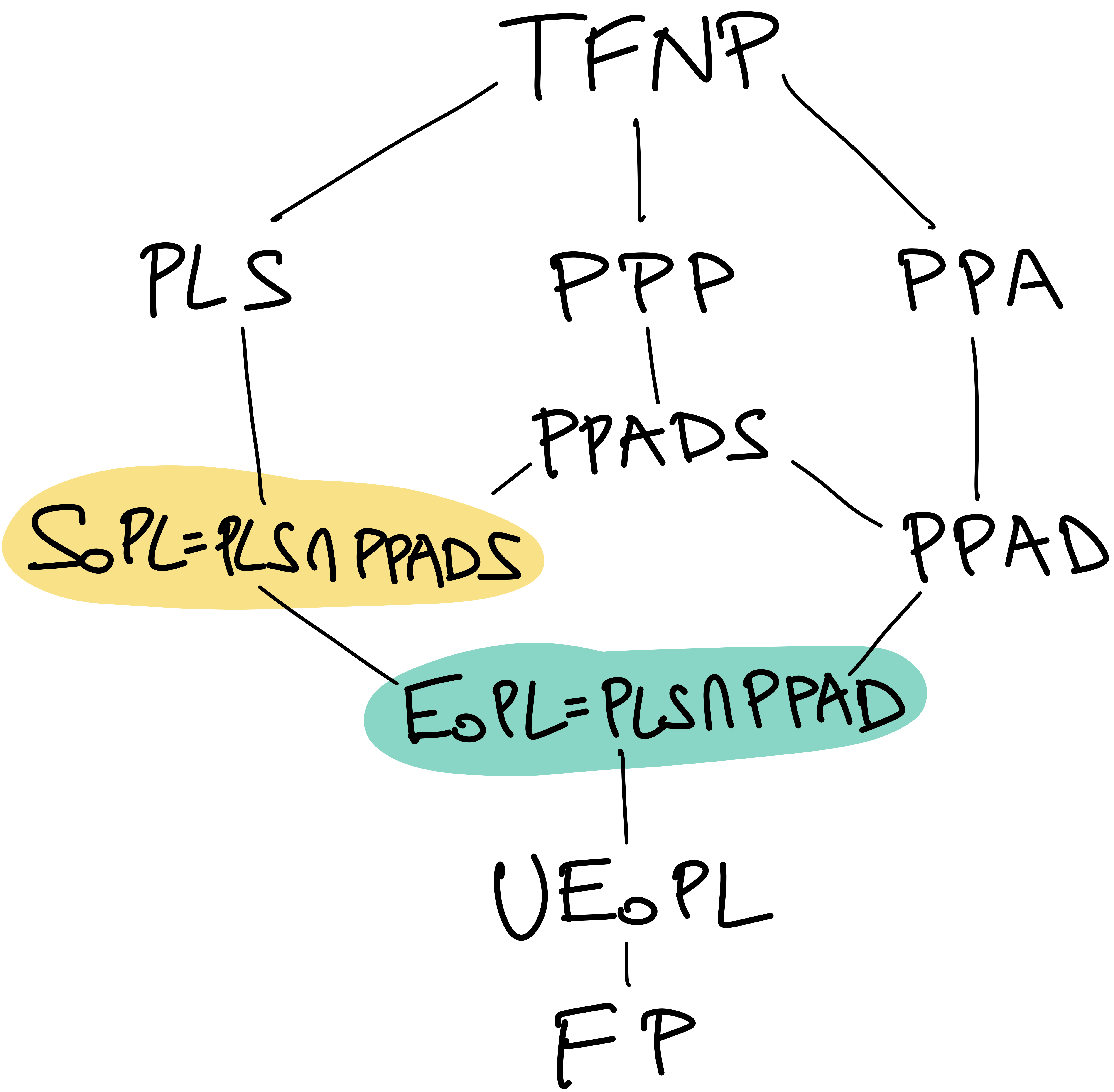
A Breakthrough Collapse (2021)



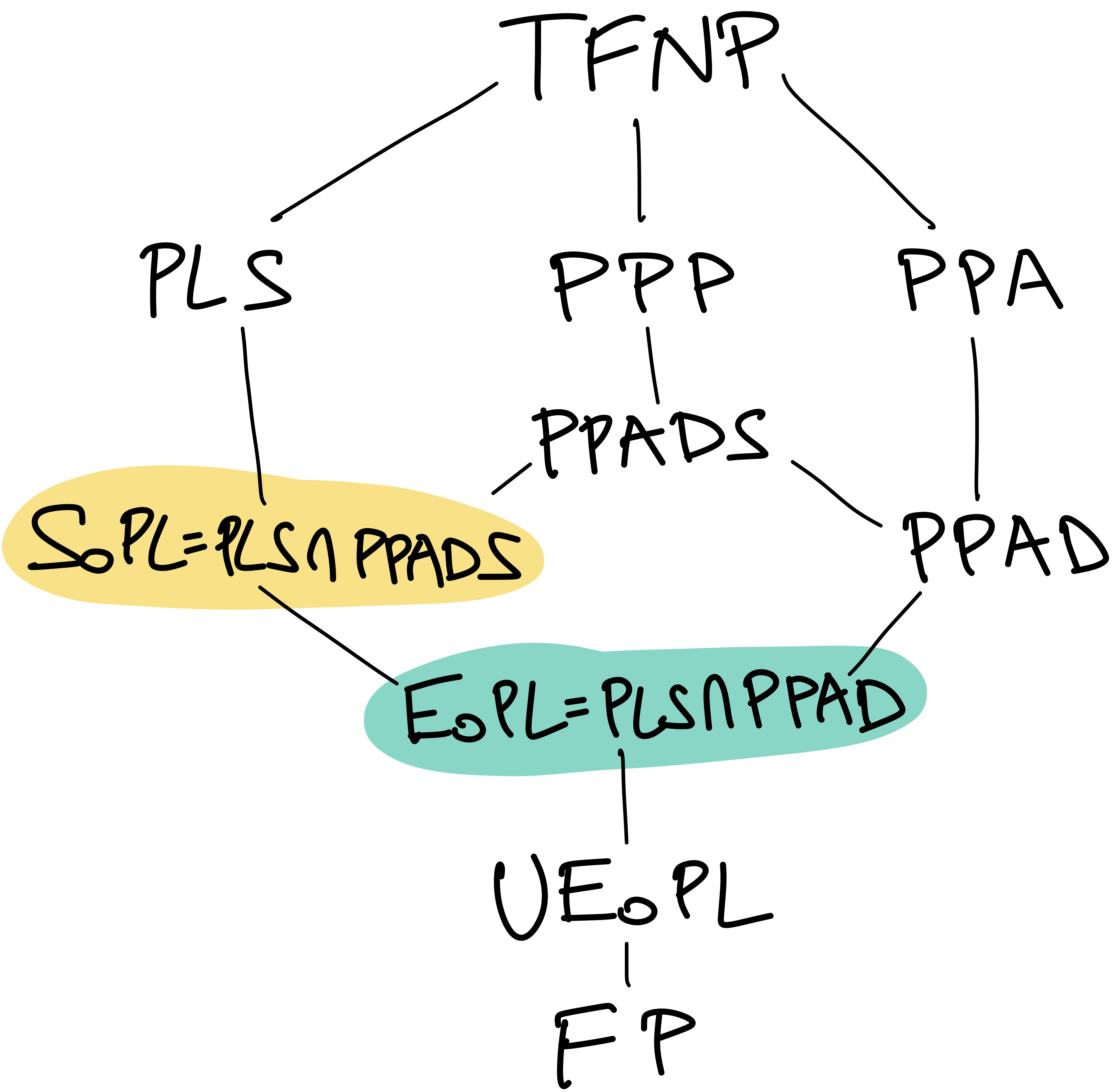
(Best paper!)
[FGHS21]

Further Collapses (2022)





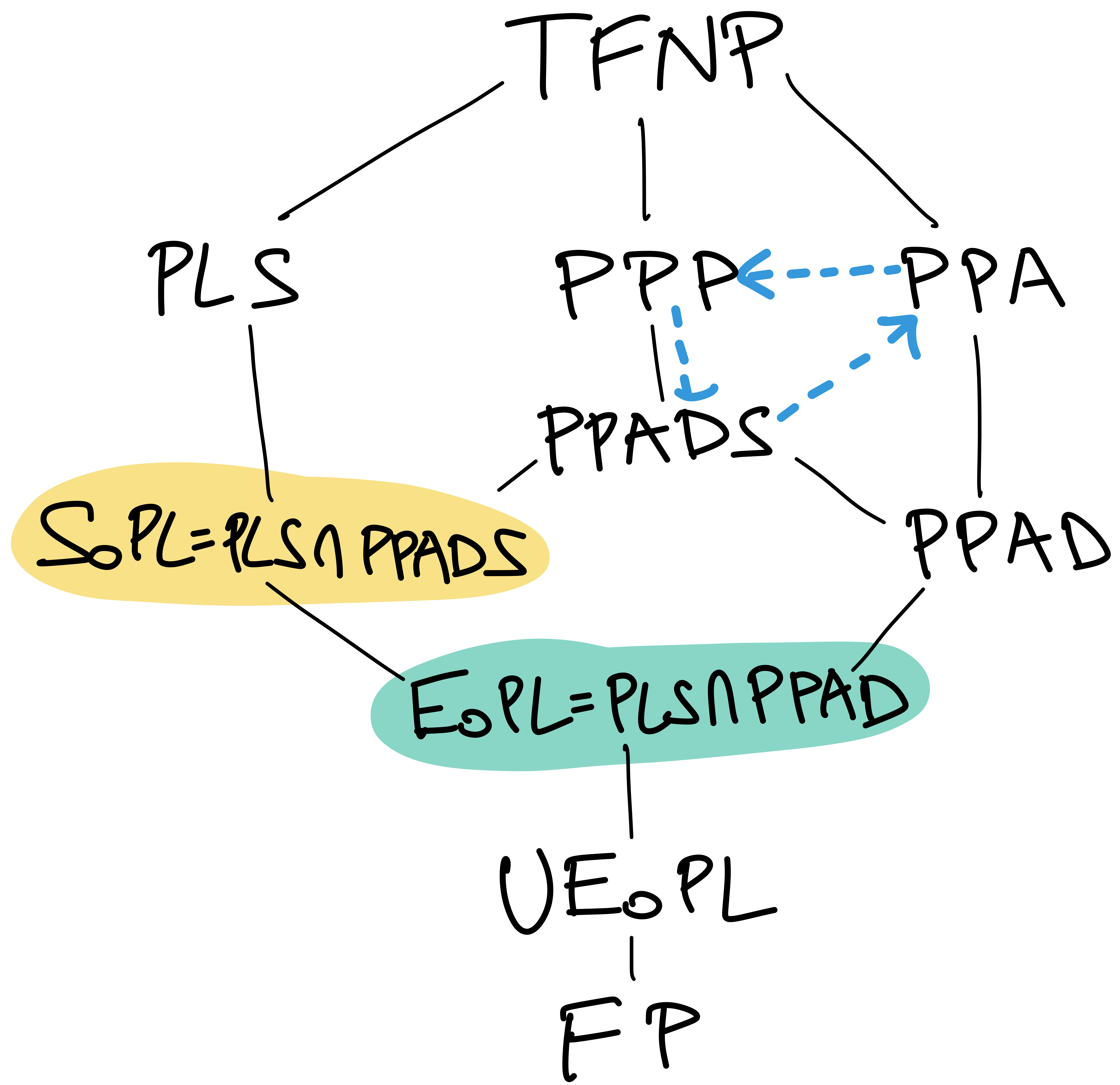
More Collapses?



More Collapses?

White-box sep. $\Rightarrow$ P $\neq$ NP

Black-box sep. possible

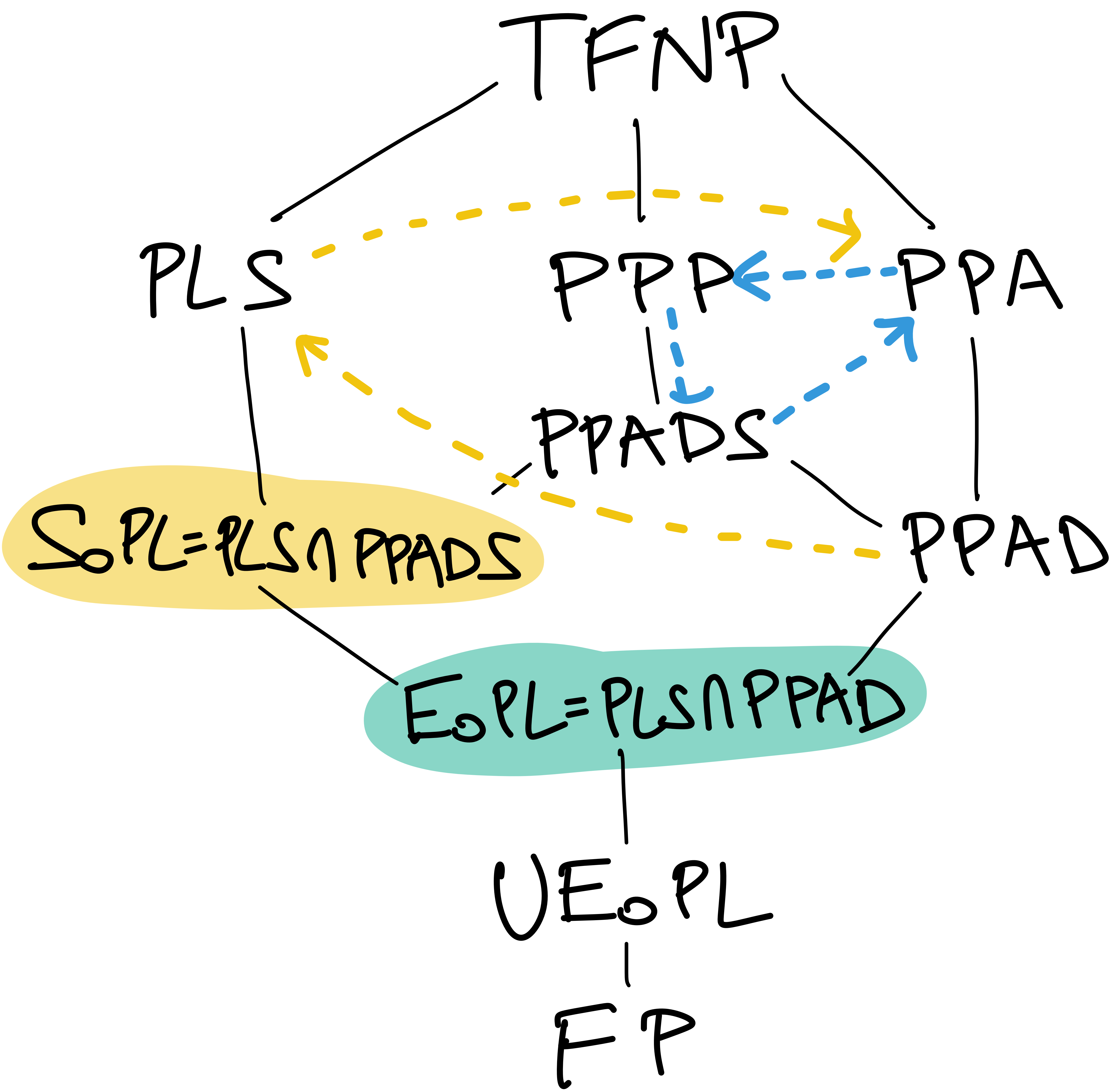


More Collapses?

White-box sep. $\Rightarrow$ $P \neq NP$

Black-box sep. possible

Beame et al. 98'



More Collapses?

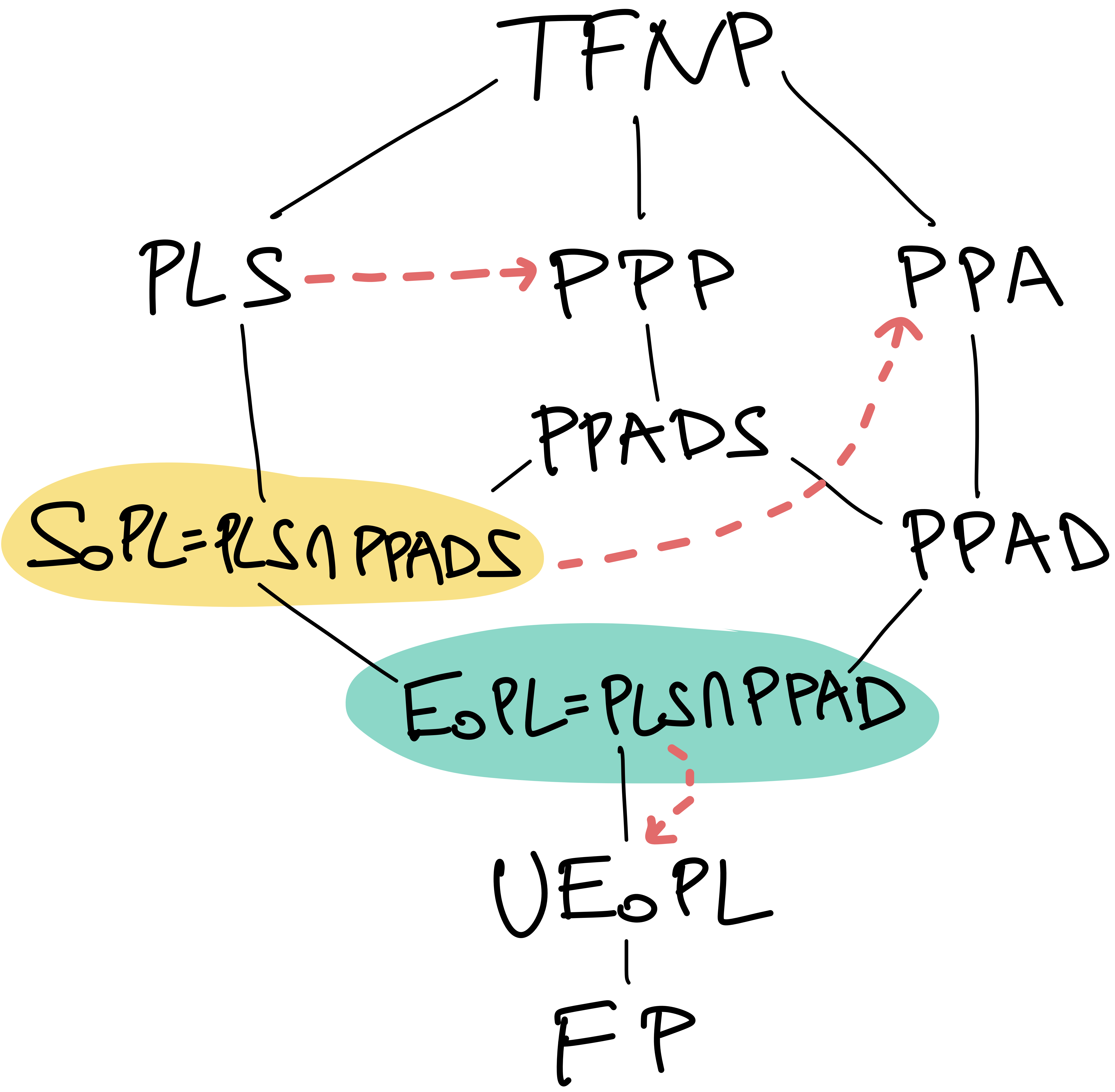
White-box sep. $\Rightarrow$ $P \neq NP$

Black-box sep. possible

Beame et al. 98'

Morioka 01'

Buresh-Openheim 04'



More Collapses?

NO MORE (BLACK-BOX)

OUR WORK

Image from https://fanart.tv/movie/9267/and-now-for-something-completely-different/

Resolution v.s. Sherali-Adams

Resolution

$$\frac{A \vee x,\ B \vee \neg x}{A \vee B}$$

measure: width

Simulated by

Sherali-Adams

$$\sum_i p_i(x) q_i(x) = 1 + J(x)$$

measure: degree

Resolution v.s. Sherali-Adams

Resolution

$$\frac{A \vee x\,,\; B \vee \neg x}{A \vee B}$$

measure: width

Simulated by

Sherali-Adams

$$\sum_i p_i(x)\,q_i(x) = 1 + J(x)$$

measure: degree

OUR RESULT: Simulation needs exp. large coefficients

Resolution v.s. Sherali-Adams

Resolution

$$\frac{A \vee x, \quad B \vee \neg x}{A \vee B}$$

measure: width

Simulated by

Sherali-Adams

$$\sum_i p_i(x) q_i(x) = 1 + J(x)$$

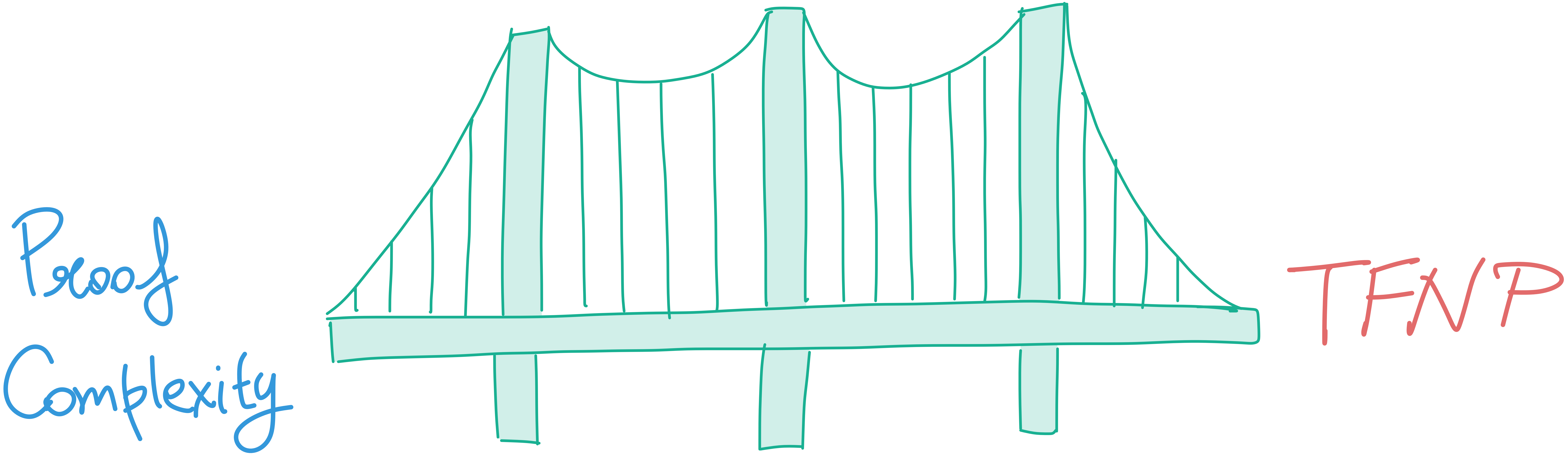
measure: degree

OUR RESULT: Simulation needs exp. large coefficients

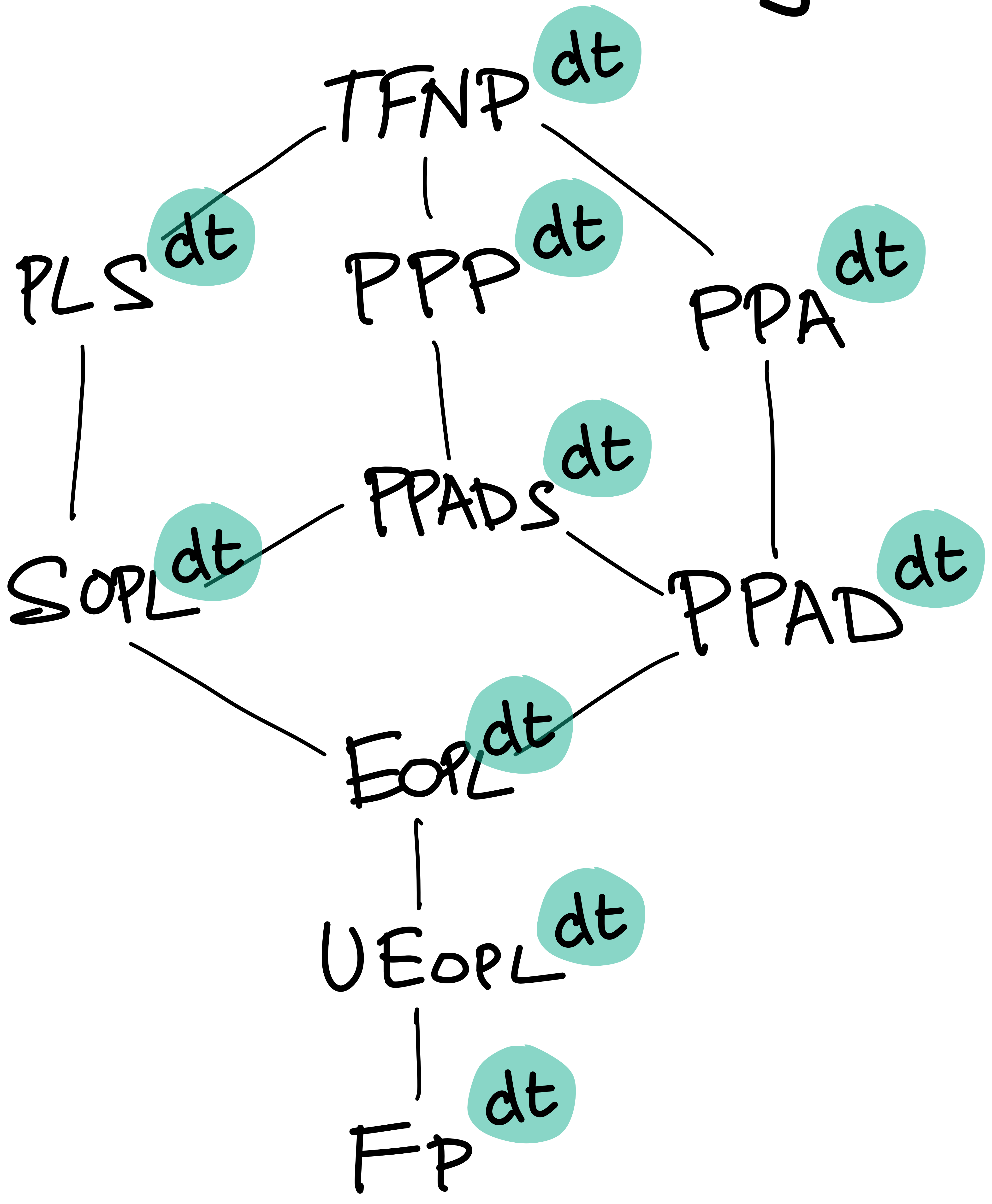
$\Updownarrow$

PLS & PPADS

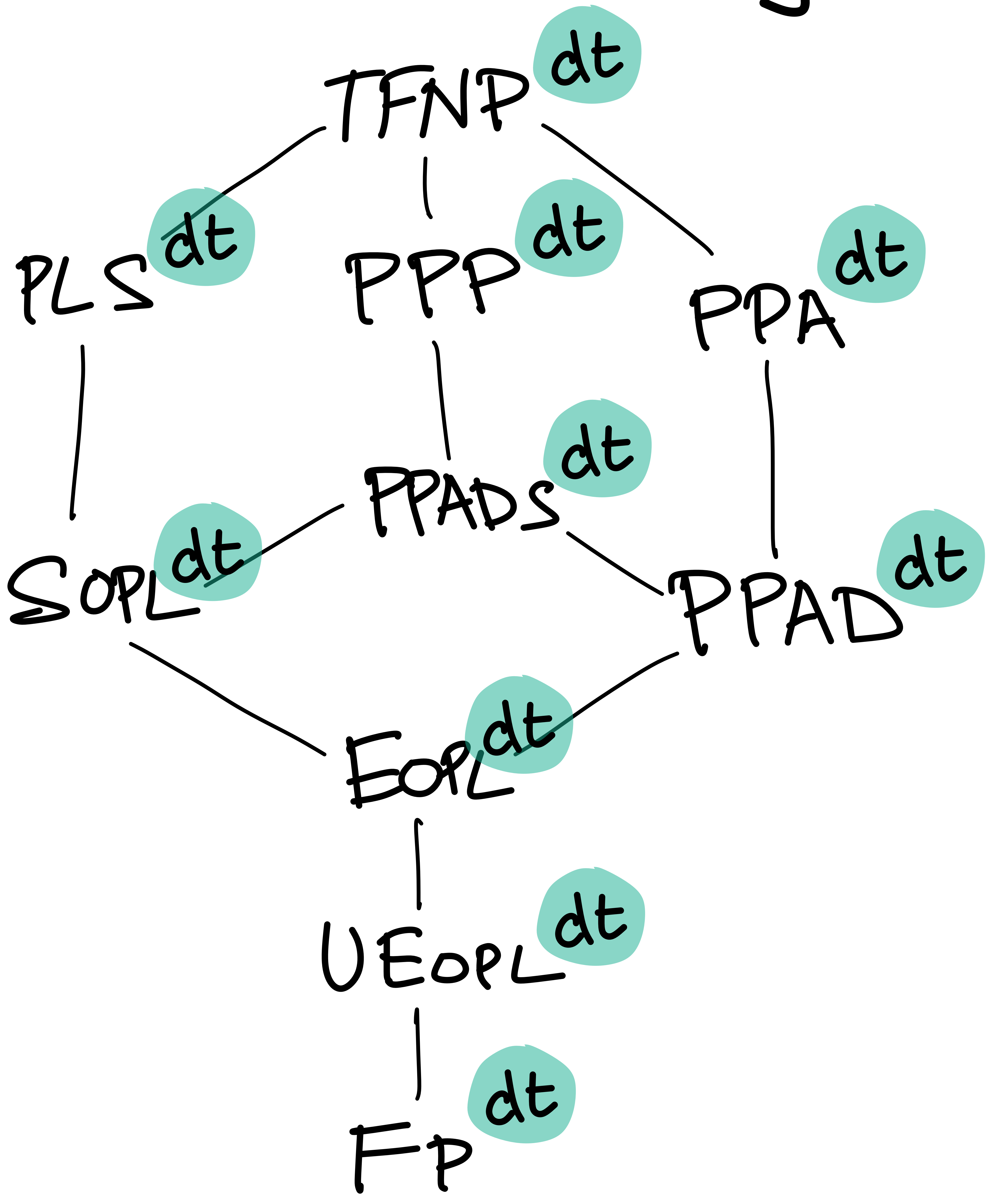
THE BRIDGE



World 1: Query analogues



World 1: Query analogues

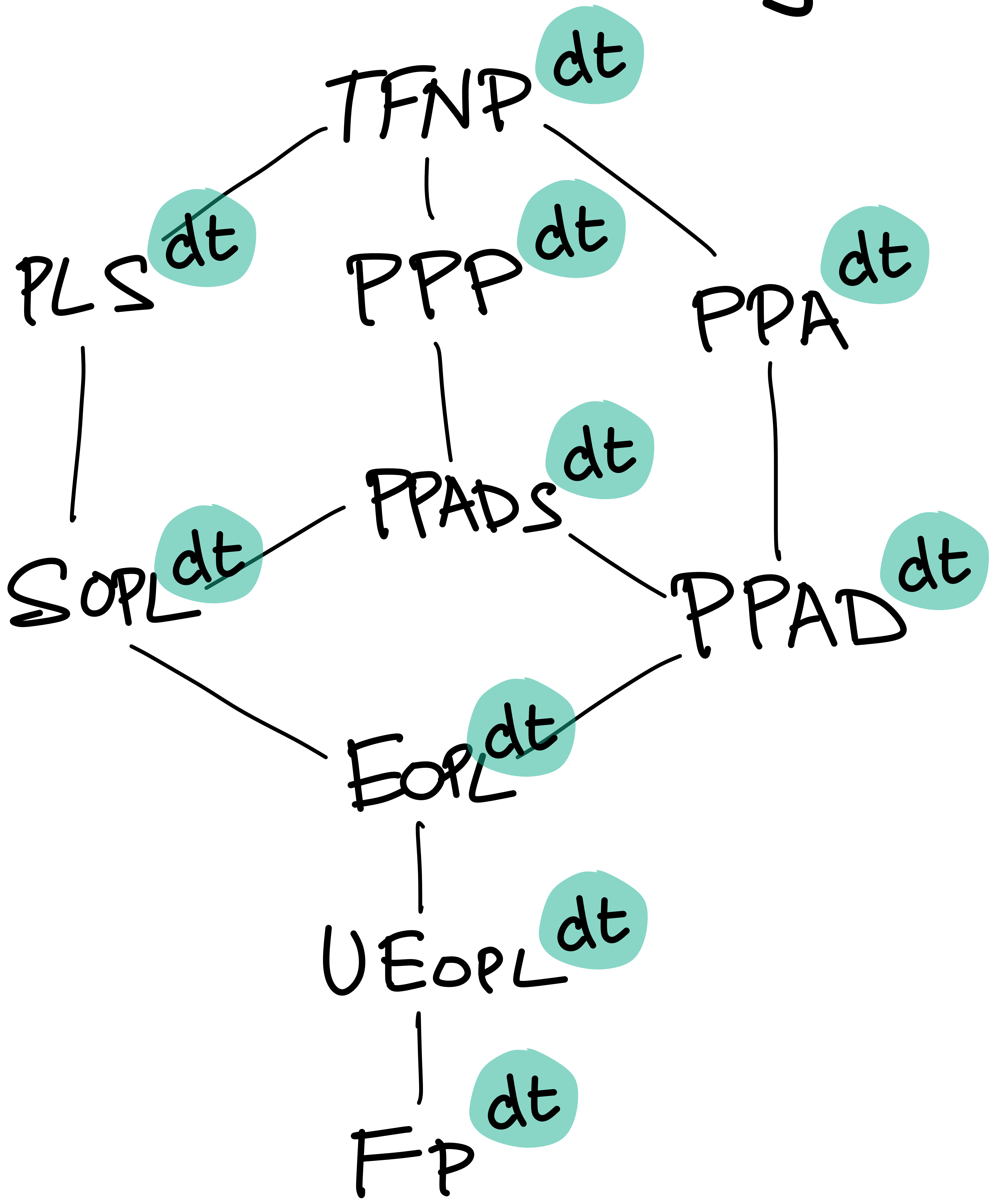


- dt

  |||

  query analogue

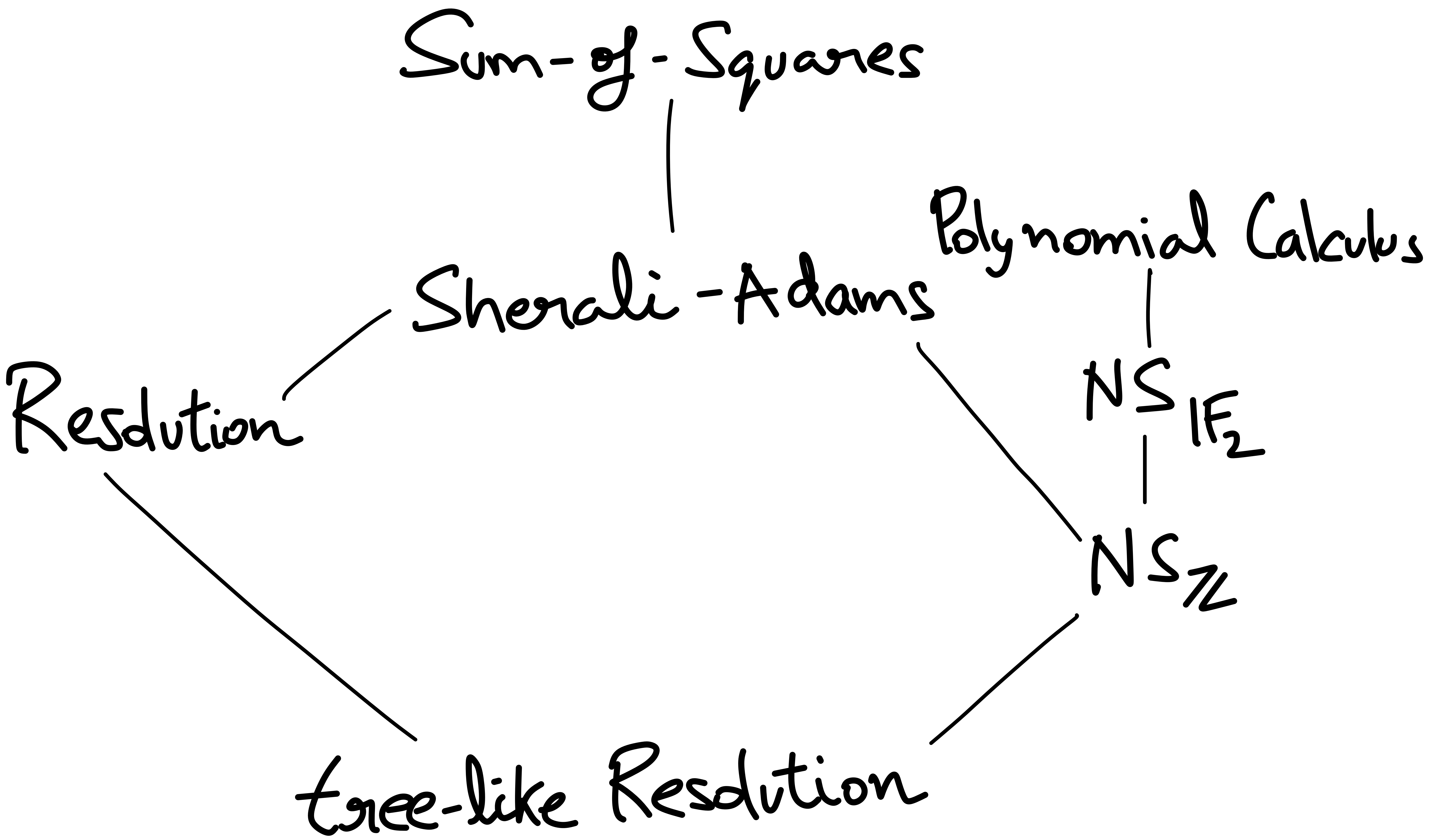
World 1: Query analogues



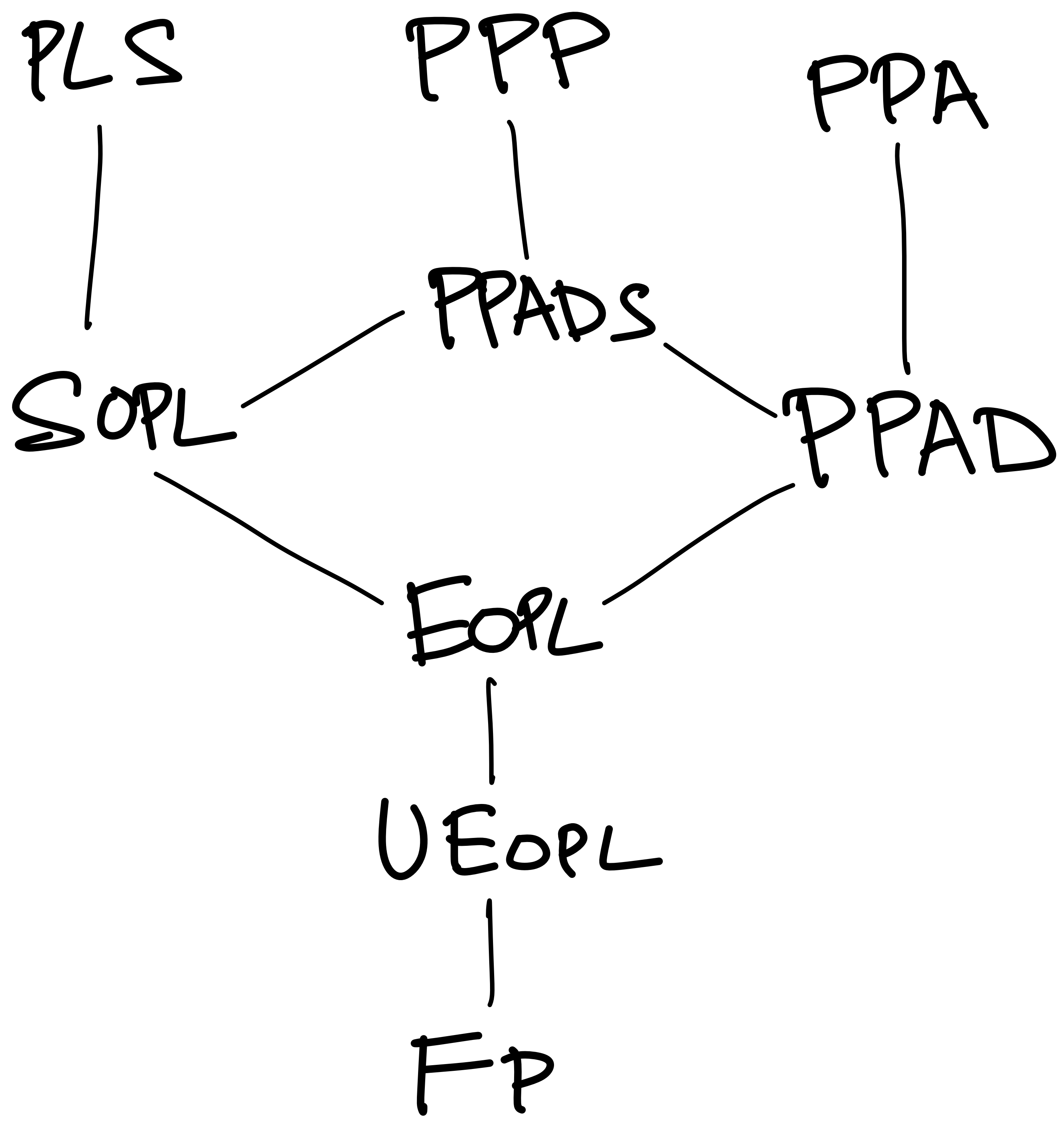
- dt ≡ query analogue
- Reductions ≡ shallow decision trees

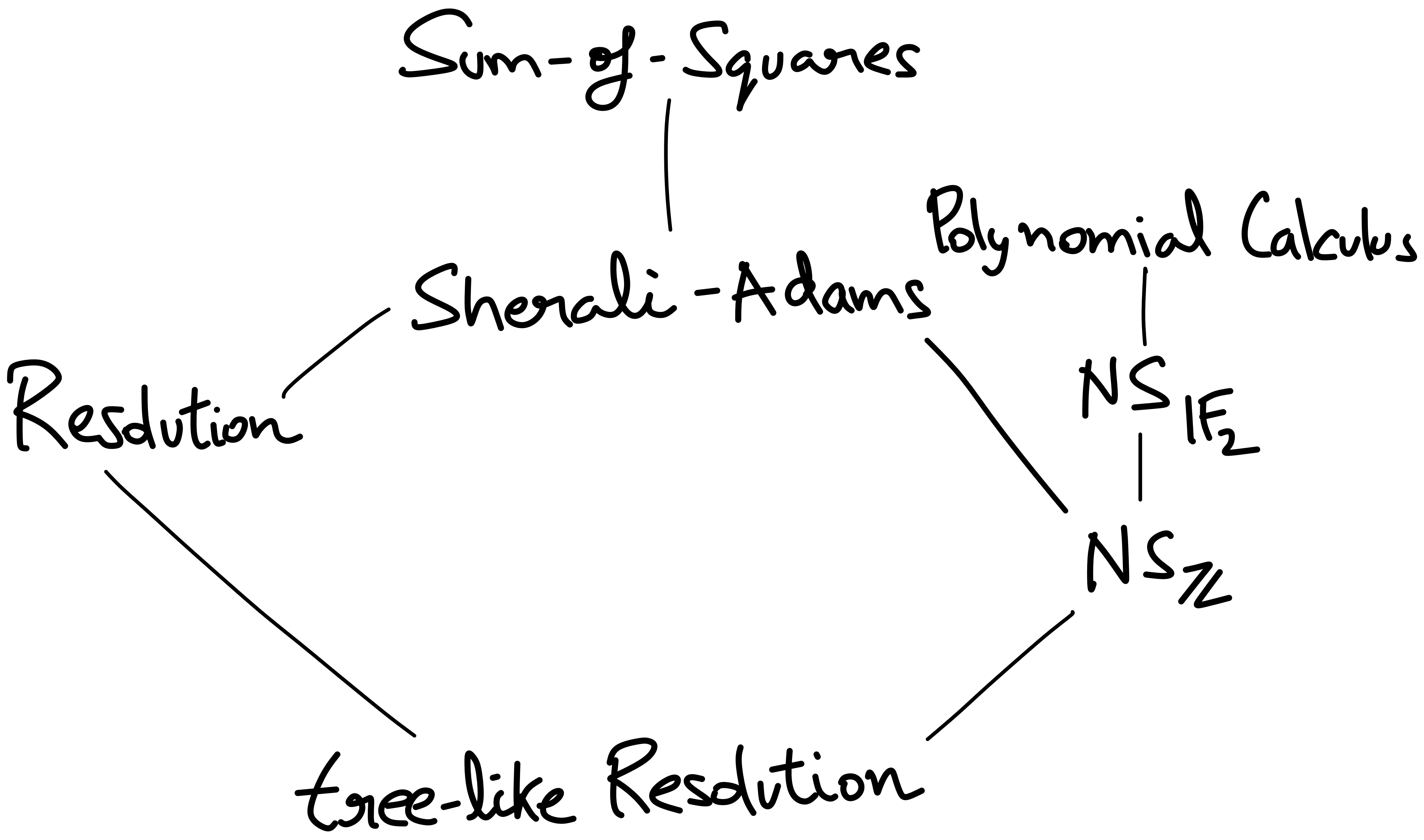
World 2: Proof Complexity

Is there a short derivation that this CNF is unsat?

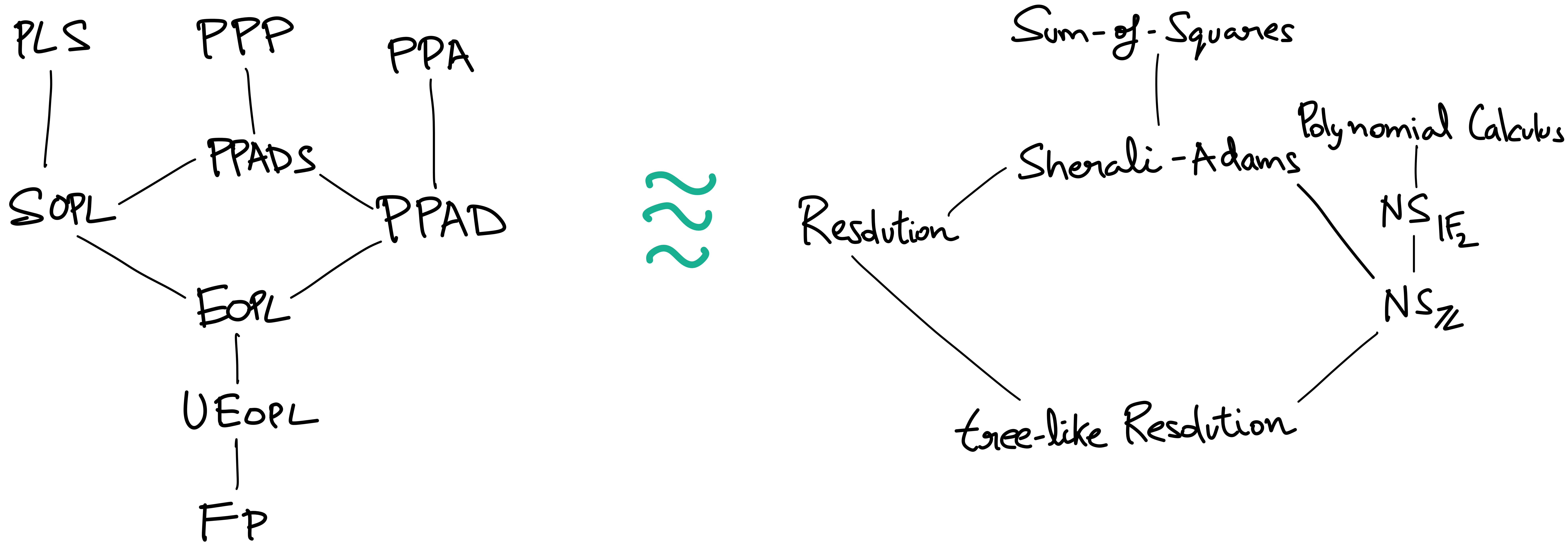


Time to Squint





Time to Squint



The Bridge: Characterizations

- TFNP^{dt} search problems can be translated into CNF fallacies

$$\text{SINK-OF-DAG} \mapsto \text{“this dag has no sinks”}$$

The Bridge: Characterizations

- TFNP^{dt} search problems can be translated into CNF fallacies

- CNF fallacies define search problems

$$\varphi = x_1 \wedge (\overline{x}_1 \vee \overline{x}_2) \wedge x_2 \longmapsto \text{find}(x_1, x_2)$$

falsified clause

More Explicitly

מכון ויצמן למדע WEIZMANN INSTITUTE OF SCIENCE

Electronic Colloquium on Computational Complexity

Under the auspices of the Computational Complexity Foundation (CCF)

Search

Login | Register | Classic Style

Submit Paper

Home | Call for Papers | Reports | Authors | Books + Lectures | About ECCC | Pointers to | Further links | Newsletter

REPORTS > DETAIL:

Revision(s):

Revision #2 to TR22-141 | 30th November 2022 03:22

Contact | Add Comment

TFNP Characterizations of Proof Systems and Monotone Circuits

PDF Revision #2

Authors: Sam Buss, Noah Fleming, Russell Impagliazzo

Accepted on: 30th November 2022 03:22
Downloads: 81
Keywords:

Abstract:

Connections between proof complexity and circuit complexity have become major tools for obtaining lower bounds in both areas. These connections -- which take the form of interpolation theorems and query-to-communication lifting theorems -- translate efficient proofs into small circuits, and vice versa, allowing tools from one area to be applied to the other. Recently, the theory of TFNP has emerged as a unifying framework underlying these connections. For many of the proof systems which admit such a connection there is a TFNP problem which characterizes it: the class of problems which are reducible to this TFNP problem via query-efficient reductions is equivalent to the tautologies that can be efficiently proven in the system. Through this, proof complexity has become a major tool for proving separations in black-box TFNP. Similarly, for certain monotone circuit models, the class of functions that it can compute efficiently is equivalent to what can be reduced to a certain TFNP problem in low communication. When a TFNP problem has both a proof and circuit characterization, one can prove an interpolation theorem. Conversely, many lifting theorems can be viewed as relating the communication and query reductions to TFNP problems. This is exciting, as it suggests that TFNP provides a roadmap for the development of further interpolation theorems and lifting theorems.

TFNP Problems

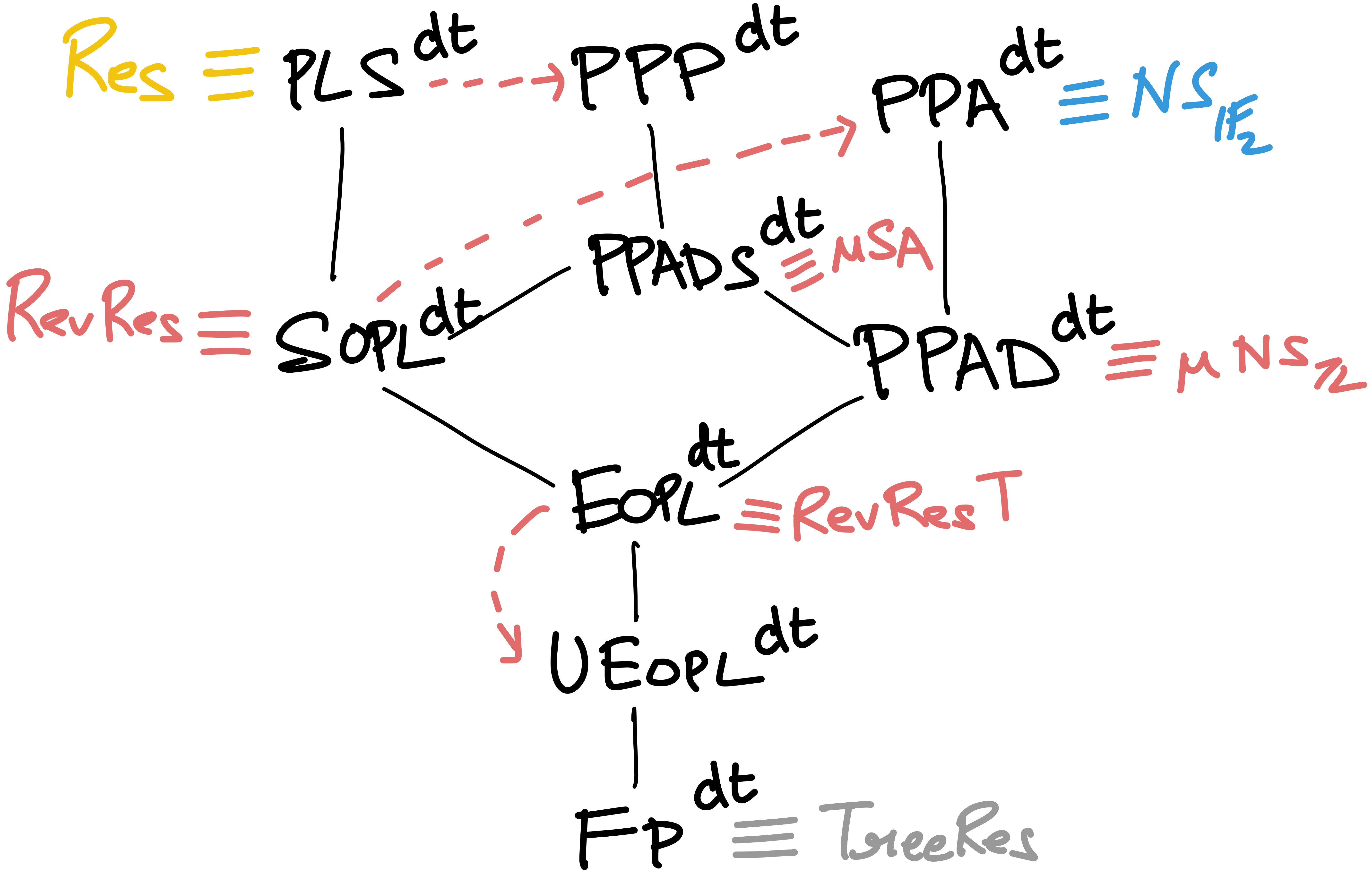
|||

Proof Systems with Reflection Principle

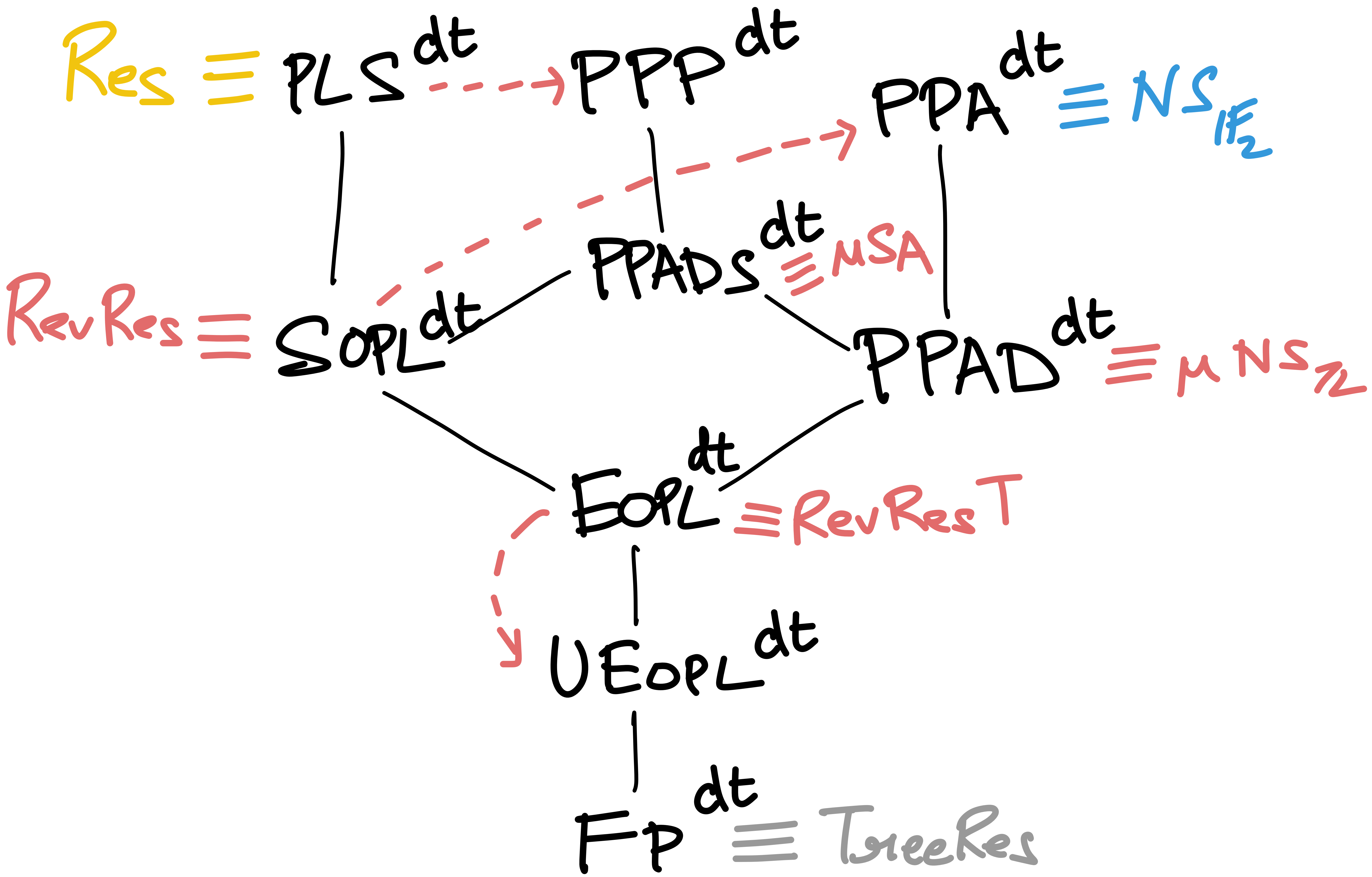
The Bridge: Characterizations

$$
\begin{tikzcd}
\mathrm{Res} \equiv \mathrm{PLS}^{dt} \arrow[dd, dash] & \mathrm{PPP}^{dt} \arrow[d, dash] & \mathrm{PPA}^{dt} \equiv \mathrm{NS}_{\mathbb{F}_2} \arrow[dd, dash] \\
& \mathrm{PPADS}^{dt} \equiv \mathrm{MSA} \arrow[dl, dash] \arrow[dr, dash] & \\
\mathrm{RevRes} \equiv \mathrm{SOPL}^{dt} \arrow[dr, dash] & & \mathrm{PPAD}^{dt} \equiv \mu\mathrm{NS}_{\mathbb{Z}} \arrow[dl, dash] \\
& \mathrm{EOPL}^{dt} \equiv \mathrm{RevResT} \arrow[d, dash] & \\
& \mathrm{UEOPL}^{dt} \arrow[d, dash] & \\
& \mathrm{FP}^{dt} \equiv \mathrm{TreeRes} &
\end{tikzcd}
$$

The Bridge : Characterizations

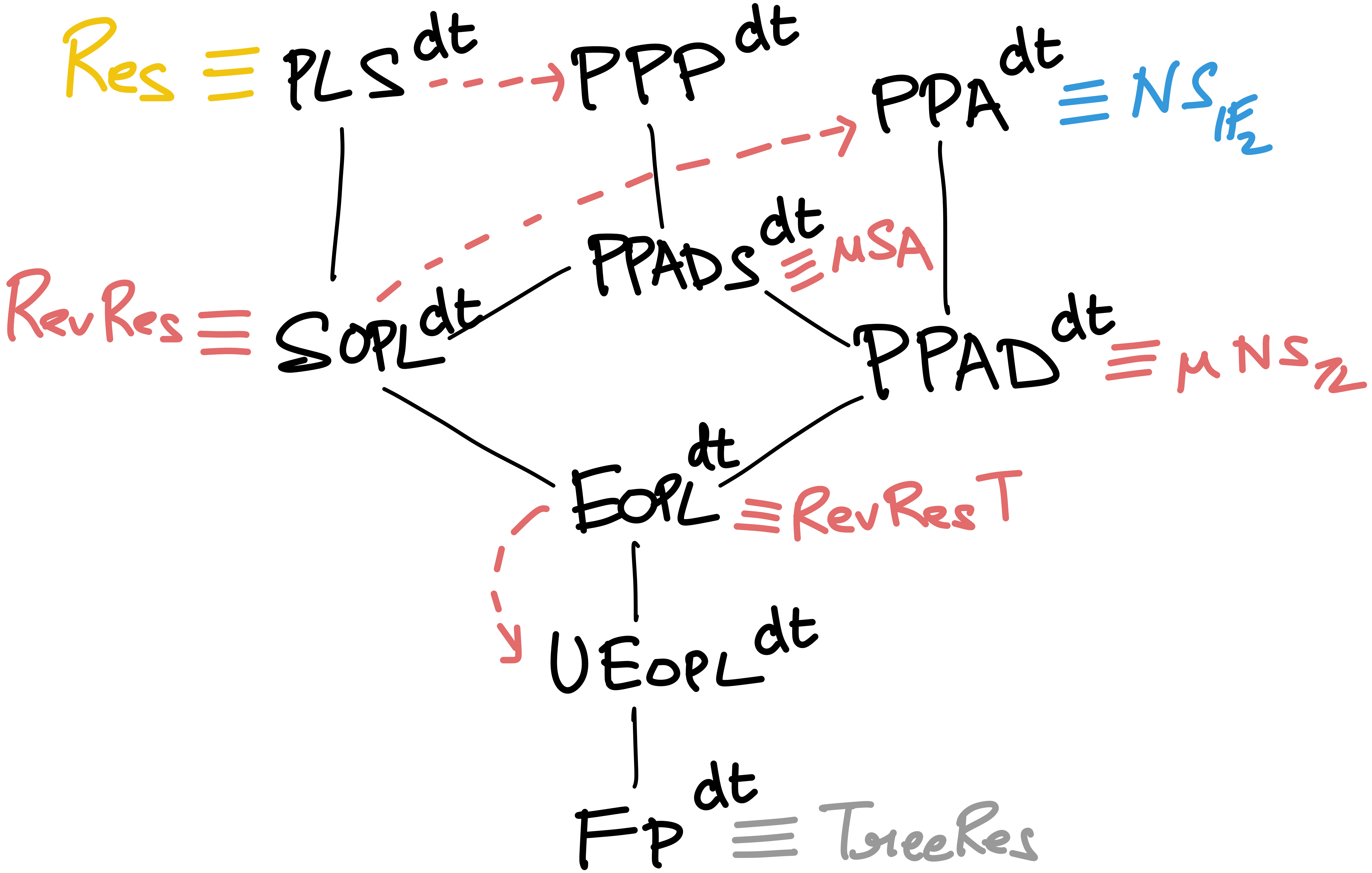


The Bridge: Characterizations



Results rephrased:

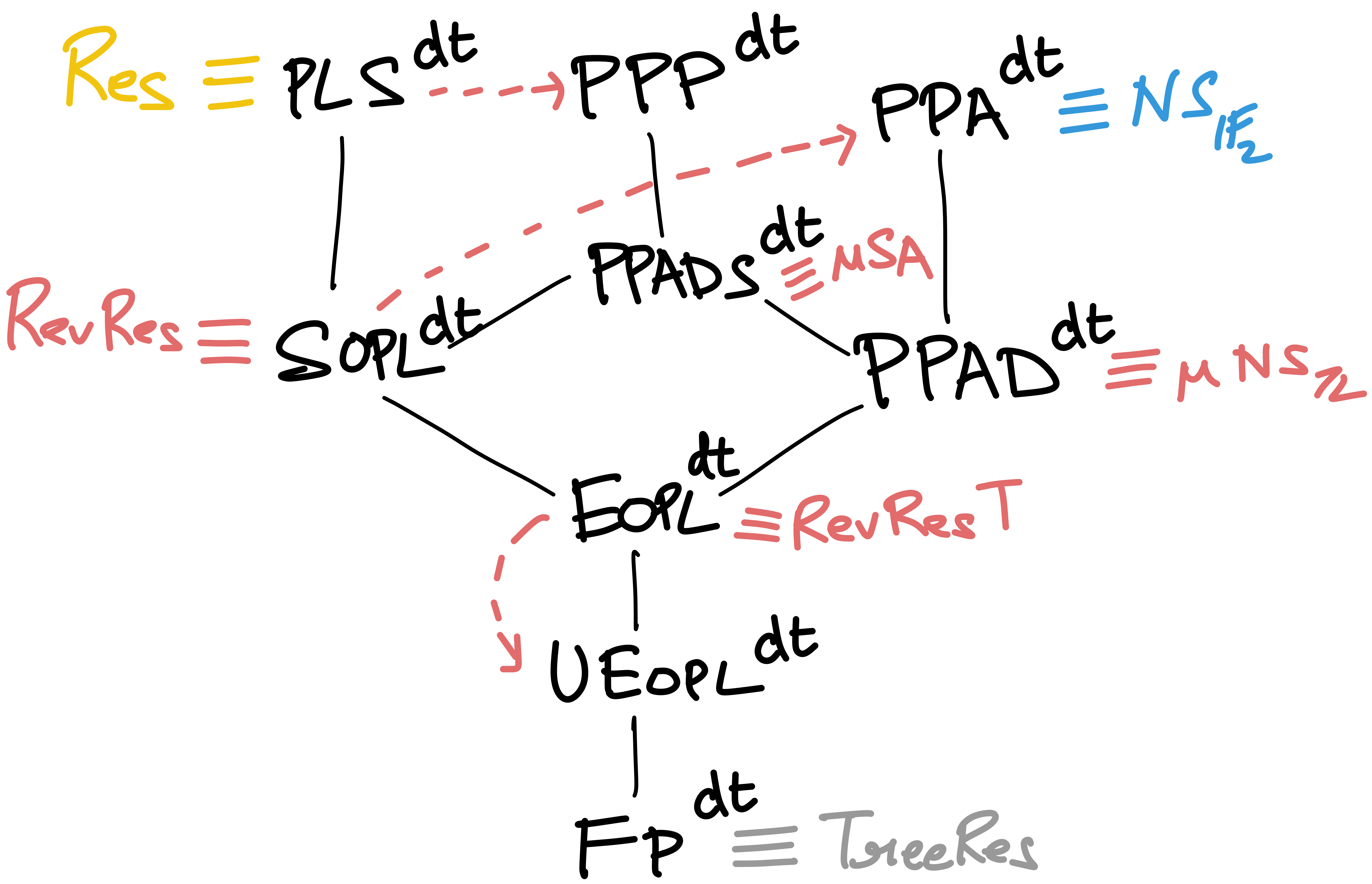
The Bridge: Characterizations



Results rephrased:

- Res $\not\leq$ uSA

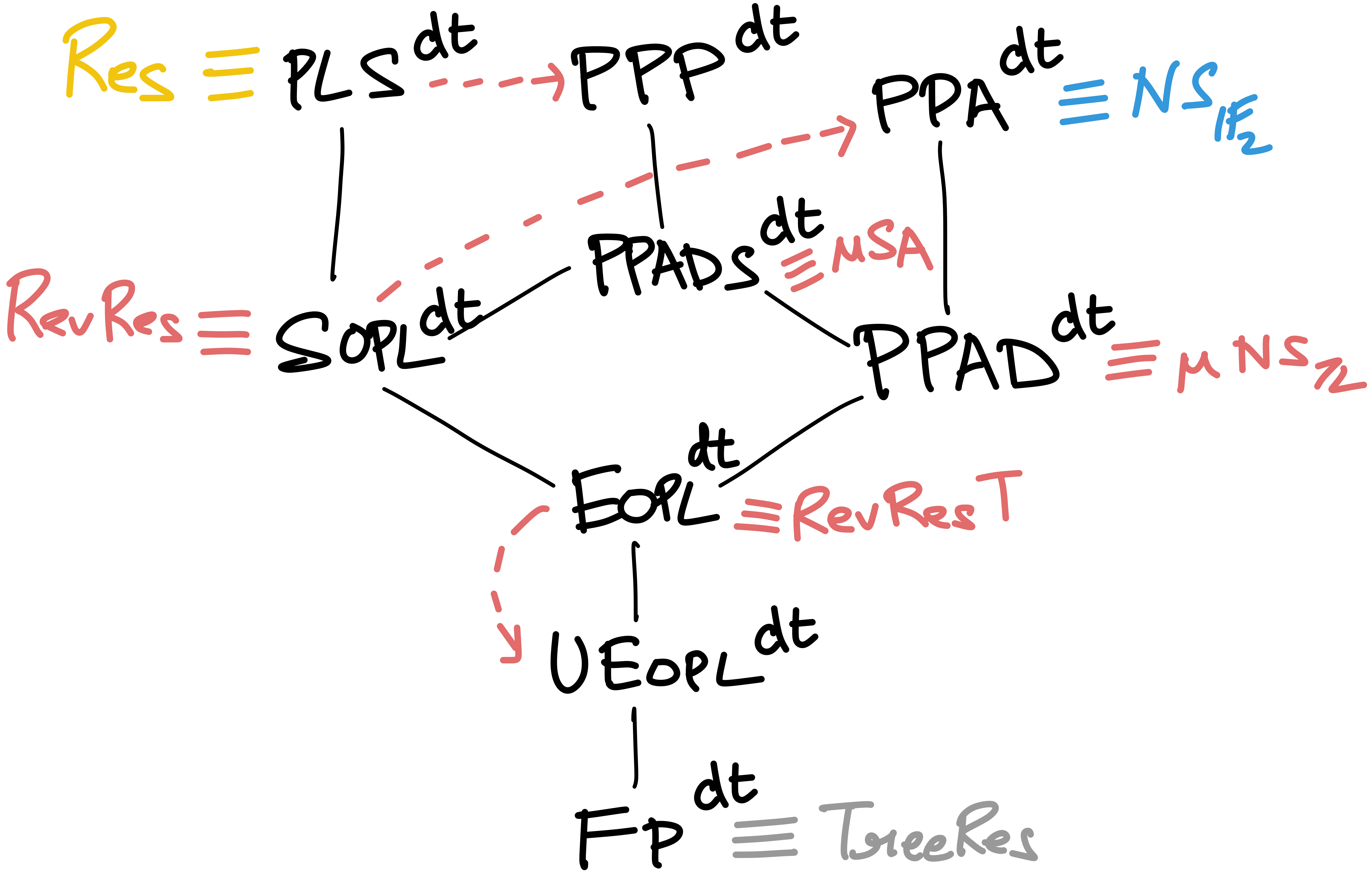
The Bridge: Characterizations



Results rephrased:

- Res $\not\subseteq$ uSA
- RevRes $\not\subseteq$ NS

The Bridge: Characterizations



Results rephrased:

- Res $\not\subseteq$ uSA
- RevRes $\not\subseteq$ NS

Independent work: $\mathrm{PLS}^{o} \not\subseteq \mathrm{PPADS}^{o} \Rightarrow \mathrm{PLS}^{o} \not\subseteq \mathrm{PPP}^{o}$ by [BT22]

Some Characterizations

Let's see why:

(i) Resolution width $\approx$ PLS^{dt} depth

Some Characterizations

Let's see why:

(i) Resolution width $\approx$ PLS^{dt} depth

Formally, given $R(x,y) \in PLS^{dt}$; PLS^{dt} depth
= depth of reduction to SoD

Some Characterizations

Let's see why:

(i) Resolution width $\approx$ PLS^{dt} depth

Formally, given $R(x,y) \in PLS^{dt}$; PLS^{dt} depth

Reduction: construct vertex set V = depth of reduction to SoD

Some Characterizations

Let's see why:

(i) Resolution width $\approx$ PLS^{dt} depth

Formally, given $R(x,y) \in PLS^{dt}$; PLS^{dt} depth

Reduction: construct vertex set V = depth of reduction to SoD

For every $v \in V$, we have decision trees

$\Pi_v(x) = s_v$

$O_v(x) = y$ s.t. $(x,y) \in R$ if v is a sink*

Some Characterizations

Let's see why:

(i) Resolution width $\approx$ PLS^{dt} depth

Formally, given $R(x,y) \in PLS^{dt}$; PLS^{dt} depth = depth of reduction to SoD

Reduction: construct vertex set V

For every $v \in V$, we have decision trees

$\Pi_v(x) = s_v$

$O_v(x) = y$ s.t. $(x,y) \in R$ if v is a sink*

$$PLS^{dt} \text{ depth} = \log |V| + \max_{v \in V} |\Pi_v|$$

Some Characterizations

Let's see why:

(i) $\boxed{a}$ Resolution width $\lesssim$ PLS^{dt} depth

Some Characterizations

Let's see why:

(i) $\boxed{a}$ Resolution width $\lesssim$ PLS^{dt} depth

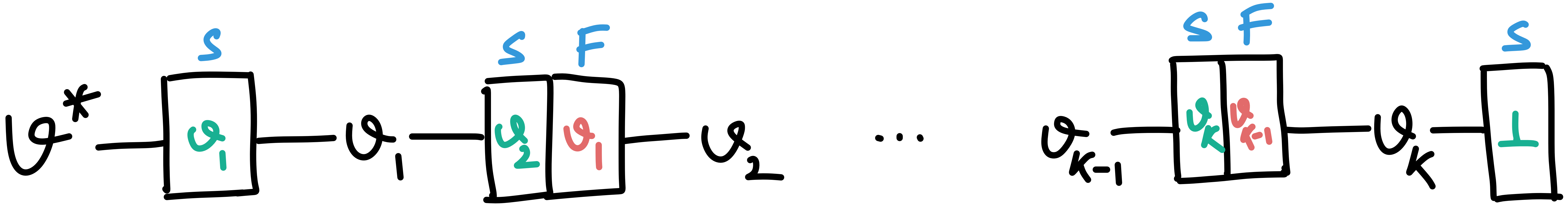
We will use Prover-Delayer characterization

Some Characterizations

Let's see why:

(i) $\boxed{a}$ Resolution width $\lesssim$ PLS^{dt} depth

We will use Prover-Delayer characterization

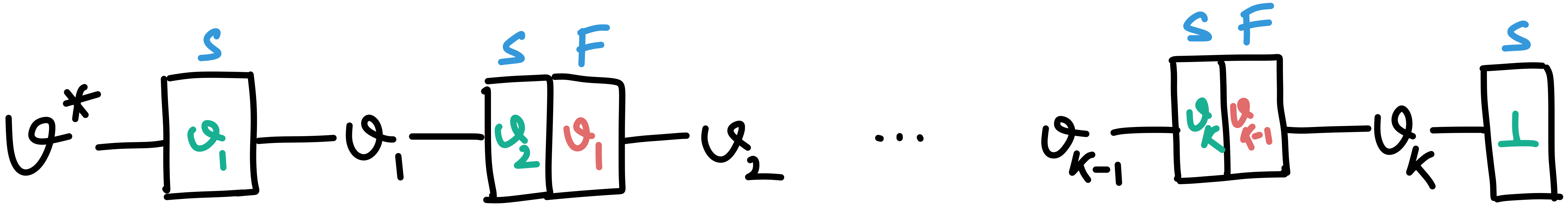


Some Characterizations

Let's see why:

(i) $\boxed{a}$ Resolution width $\lesssim$ PLS^{dt} depth

We will use Prover-Delayer characterization



$$\text{Res Width} \leq 2\log|V| + \max_{v\in V} |\Pi_v| \leq 2\cdot \text{PLS}^{dt}\text{ depth}$$

Some Characterizations

Let's see why:

(i) $\boxed{b}$ Resolution width $\gtrsim$ PLS^{dt} depth

Some Characterizations

Let's see why:

(i) $\boxed{b}$ Resolution width $\gtrsim$ PLS^{dt} depth

Dag-like Res proof is already kinda a SoD reduction

Some Characterizations

Let's see why:

(i) $\boxed{b}$ Resolution width $\gtrsim$ PLS^{dt} depth

Dag-like Res proof is already kinda a SoD reduction

Just have to Flip The Proof!

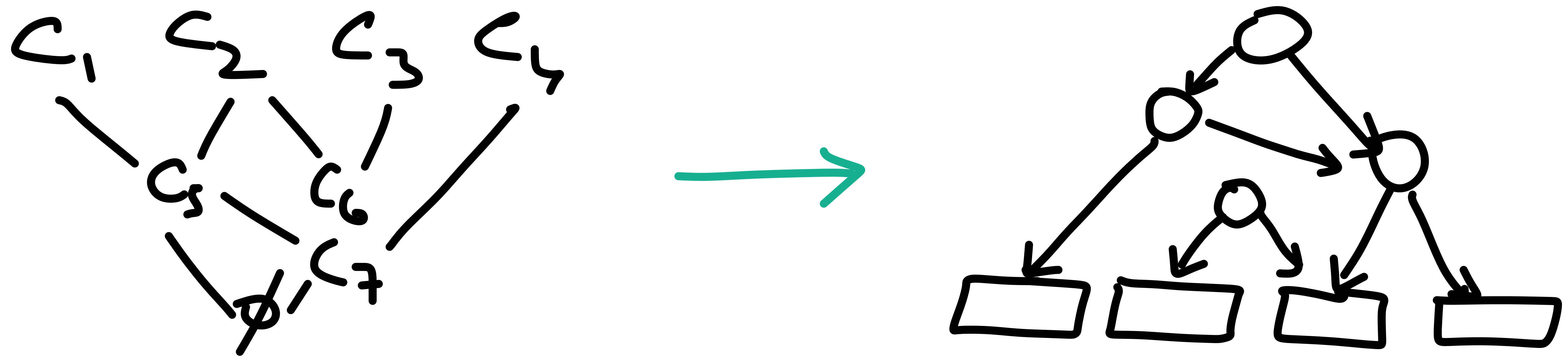
Some Characterizations

Let's see why:

(i) $\boxed{b}$ Resolution width $\gtrsim PLS^{dt}$ depth

Dag-like Res proof is already kinda a SoD reduction

Just have to Flip The Proof!



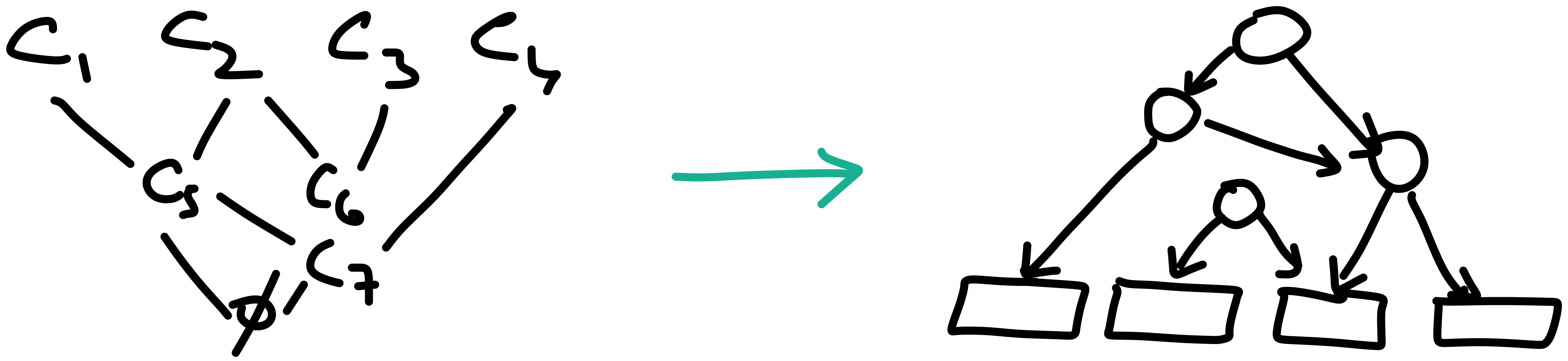
Some Characterizations

Let's see why:

(i) $\boxed{b}$ Resolution width $\gtrsim$ PLS^{dt} depth

Dag-like Res proof is already kinda a SoD reduction

Just have to Flip The Proof!



$$PLS^{dt}\ \text{depth} \leq \log(\text{Res Size}) + \text{Res Width}$$

Some Characterizations

Let's see why:

(i) Resolution width $\approx$ PLS^{dt} depth

(ii) Unary NS deg $\approx$ $PPAD^{dt}$ depth

Lemma: If $\exists$ depth d EoL-formulation of F
then $\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Lemma: If $\exists$ depth d EoL-formulation of F

then $\exists$ uNS refutation of F

with degree $O(d)$ and size $L2^{O(d)}$.

Proof:

Lemma: If $\exists$ depth d EoL-formulation of F
then $\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation: $(V=[L], \{s_v, p_v, o_v\})$

Lemma: If $\exists$ depth d EoL-formulation of F
then $\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation: $\left(V=[L], \{s_v, p_v, o_v\}\right)$

Define $$S_v(x) = \begin{cases} -1 & \text{if } v \neq v^* \text{ is a source in } G_x \\ 1 & \text{if } v \neq v^* \text{ is a proper sink in } G_x \\ 0 & \text{otherwise} \end{cases}$$

Lemma: If $\exists$ depth d EoL-formulation of F
then $\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation: $(V=[L], \{s_v, p_v, o_v\})$

Define $S_v(x) = \begin{cases} -1 & \text{if } v \neq v^* \text{ is a source in } G_x \\ 1 & \text{if } v \neq v^* \text{ is a proper sink in } G_x \\ 0 & \text{otherwise} \end{cases}$

S_v can be computed in depth $5d$.

Lemma: If $\exists$ depth d EoL-formulation of F
then $\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation: $(V=[L], \{s_v, p_v, o_v\})$

Define $S_v(x) = \begin{cases} -1 & \text{if } v \neq v^* \text{ is a source in } G_x \\ 1 & \text{if } v \neq v^* \text{ is a proper sink in } G_x \\ 0 & \text{otherwise} \end{cases}$

S_v can be computed in depth $5d$.

$$S_v = \sum_{(-1)\text{-leaf } \ell} -D_\ell + \sum_{1\text{-leaf } \ell} D_\ell$$

Lemma: If $\exists$ depth d EoL-formulation of F
then $\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation: $(V=[L], \{s_v, p_v, o_v\})$

Define $S_v(x) = \begin{cases} -1 & \text{if } v \neq v^* \text{ is a source in } G_x \\ 1 & \text{if } v \neq v^* \text{ is a proper sink in } G_x \\ 0 & \text{otherwise} \end{cases}$

S_v can be computed in depth $5d$.

$$S_v = \sum_{(-1)\text{-leaf } \ell} -D_\ell + \sum_{1\text{-leaf } \ell} D_\ell = \sum_{(-1)\text{-leaf } \ell} -D'_\ell \bar{C}_\ell + \sum_{1\text{-leaf } \ell} D'_\ell \bar{C}_\ell$$

Lemma: If $\exists$ depth d EoL-formulation of F

then $\exists$ uNS refutation of F with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation: $(V=[L], \{s_v, p_v, o_v\})$

Define $$S_v(x) = \begin{cases} -1 & \text{if } v \neq v^* \text{ is a source in } G_x \\ 1 & \text{if } v \neq v^* \text{ is a proper sink in } G_x \\ 0 & \text{otherwise} \end{cases}$$

S_v can be computed in depth $5d$.

$$S_v = \sum_{(-1)\text{-leaf } \ell} -D_\ell + \sum_{1\text{-leaf } \ell} D_\ell = \sum_{(-1)\text{-leaf } \ell} -D'_\ell \overline{C_\ell} + \sum_{1\text{-leaf } \ell} D'_\ell \overline{C_\ell}$$

leaves are solutions

Lemma then: If $\exists$ depth d EoL-formulation of F
$\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation : $(V=[L], \{s_v, p_v, o_v\})$

Define $$S_v(x) = \begin{cases} -1 & \text{if } v \neq v^* \text{ is a source in } G_x \\ 1 & \text{if } v \neq v^* \text{ is a proper sink in } G_x \\ 0 & \text{otherwise} \end{cases}$$

S_v can be computed in depth $5d$.

$$S_v = \sum_{(-1)\text{-leaf } \ell} -D_\ell + \sum_{1\text{-leaf } \ell} D_\ell = \sum_{(-1)\text{-leaf } \ell} -D'_\ell \overline{C_\ell} + \sum_{1\text{-leaf } \ell} D'_\ell \overline{C_\ell}$$

leaves are solutions

$$\Rightarrow \sum_{v \in V} S_v = \sum_i p_i \overline{C_i}$$

for some $\{p_i\}$

Lemma: If $\exists$ depth d EoL-formulation of F
then $\exists$ uNS refutation of F
with degree $O(d)$ and size $L2^{O(d)}$.

Proof: EoL formulation: $(V=[L], \{s_v, p_v, o_v\})$

Define $S_v(x) = \begin{cases} -1 & \text{if } v \neq v^* \text{ is a source in } G_x \\ 1 & \text{if } v \neq v^* \text{ is a proper sink in } G_x \\ 0 & \text{otherwise} \end{cases}$

S_v can be computed in depth $5d$.

$$S_v = \sum_{(-1)\text{-leaf } \ell} -D_\ell + \sum_{1\text{-leaf } \ell} D_\ell = \sum_{(-1)\text{-leaf } \ell} -D'_\ell \overline{C}_\ell + \sum_{1\text{-leaf } \ell} D'_\ell \overline{C}_\ell$$

leaves are solutions

$$\Rightarrow \sum_{v \in V} S_v = \sum_i p_i \overline{C}_i = \#\text{sinks in } G_x - \#\text{ non-}v^*\text{ sources in } G_x = 1$$

for some $\{p_i\}$

Lemma: If $\exists$ uNS refutation of F with deg d and size L
then $\exists$ deg $O(d)$ $EoL_{O(L)}$-formulation of F.

Lemma: If $\exists$ uNS refutation of F with deg d and size L
then $\exists$ deg $O(d)$ $EoL_{O(L)}$-formulation of F.

Proof:

Lemma: If $\exists$ uNS refutation of F with deg d and size L
then $\exists$ deg $O(d)$ $EoL_{O(L)}$-formulation of F.

Proof: The refutation: $\sum_{i=1}^{m} p_i \overline{C_i} = 1 = \sum_i \sum_j c_{ij} q_{ij}$

Lemma: If $\exists$ uNS refutation of F with deg d and size L
then $\exists$ deg $O(d)$ $EoL_{O(L)}$-formulation of F.

Proof: The refutation: $\sum_{i=1}^{m} p_i \overline{C_i} = 1 = \sum_i \sum_j c_{ij} q_{ij}$

In EoL formulation,

Lemma: If $\exists$ uNS refutation of F with deg d and size L
then $\exists$ deg $O(d)$ $EoL_{O(L)}$-formulation of F.

Proof: The refutation: $\sum_{i=1}^{m} p_i \overline{C_i} = 1 = \sum_i \sum_j c_{ij} q_{ij}$

In EoL formulation,

Nodes $\equiv q_{ij}$ with multiplicity $|c_{ij}| \longrightarrow$ in $+,-$ set based on $\mathrm{sgn}(c_{ij})$

$+ \; V^* = \{v^*\} \subseteq -$

Lemma: If $\exists$ uNS refutation of F with deg d and size L
then $\exists$ deg $O(d)$ $EoL_{O(L)}$-formulation of F.

Proof: The refutation: $\sum_{i=1}^{m} p_i \overline{C_i} = 1 = \sum_i \sum_j c_{ij} q_{ij}$

In EoL formulation,

Nodes $\equiv$ q_{ij} with multiplicity $|c_{ij}|$ $\longrightarrow$ in $+,-$ set based on $\mathrm{sgn}(c_{ij})$

$+ \; V^* = \{v^*\} \subseteq -$

Edges $\equiv$ Outer Matching + Inner Matching

Lemma: If $\exists$ uNS refutation of F with deg d and size L then $\exists$ deg $O(d)$ $EoL_{O(L)}$-formulation of F.

Proof: The refutation: $\sum_{i=1}^{m} p_i \bar{C}_i = 1 = \sum_i \sum_j c_{ij} q_{ij}$

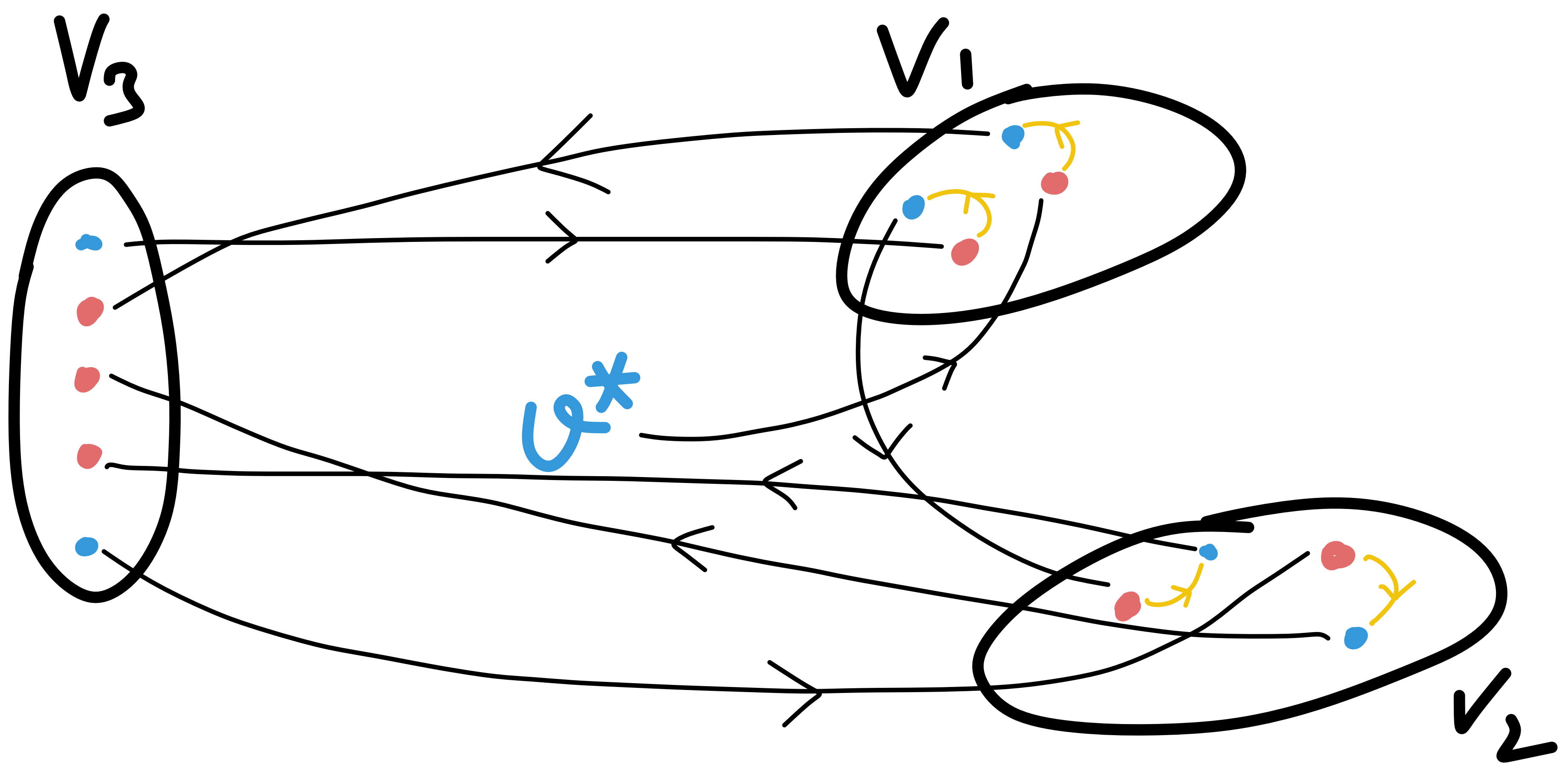
In EoL formulation,

Nodes $\equiv q_{ij}$ with multiplicity $|c_{ij}| \longrightarrow$ in $+,-$ set based on $\mathrm{sgn}(c_{ij})$

$+ \; V^* = \{v^*\} \subseteq -$

Edges $\equiv$ Outer Matching + Inner Matching

LEGEND
- $v \in +$
- $v \in -$
- fixed edge
- variable edge



On Separations

On Separations

Key Lemma:

On Separations

Key Lemma : Robust separation of SoPL from NS

SoPL ≈ SoD without merging of paths

On Separations

Key Lemma : Robust separation of SoPL from NS

SoPL ≈ SoD without merging of paths

Robust? We modify NS to refute approximately

On Separations

Key Lemma: Robust separation of SoPL from NS

SoPL ≈ SoD without merging of paths

Robust? We modify NS to refute approximately

$$\varepsilon\text{-NS} := \sum_{i \in [m]} p_i(x) \cdot a_i(x) = 1 \pm \varepsilon \quad \forall x \in \{0,1\}^n$$

On Separations

Key Lemma: Robust separation of SoPL from NS

SoPL ≈ SoD without merging of paths

Robust? We modify NS to refute approximately

$$\varepsilon\text{-NS} := \sum_{i \in [m]} p_i(x) \cdot a_i(x) = 1 \pm \varepsilon \quad \forall x \in \{0,1\}^n$$

NOTE: Not a Cook-Reckhow proof system!

Verification is coNP-complete.

Lemma: Every $\frac{1}{2}$-NS refutation of $SoPL_n$ requires deg $n^{\Omega(1)}$.

Lemma: Every $\frac{1}{2}$-NS refutation of $SoPL_n$ requires deg $n^{\Omega(1)}$.

IDEA: Randomized decision-to-search reduction in the style of Raz-Wigderson '92 (matching)

Lemma: Every $\frac{1}{2}$-NS refutation of $SoPL_n$ requires deg $n^{\Omega(1)}$.

IDEA: Randomized decision-to-search reduction in the style of Raz-Wigderson '92 (matching)

We show ε-NS proofs for SoPL $\Rightarrow$ apx poly for OR

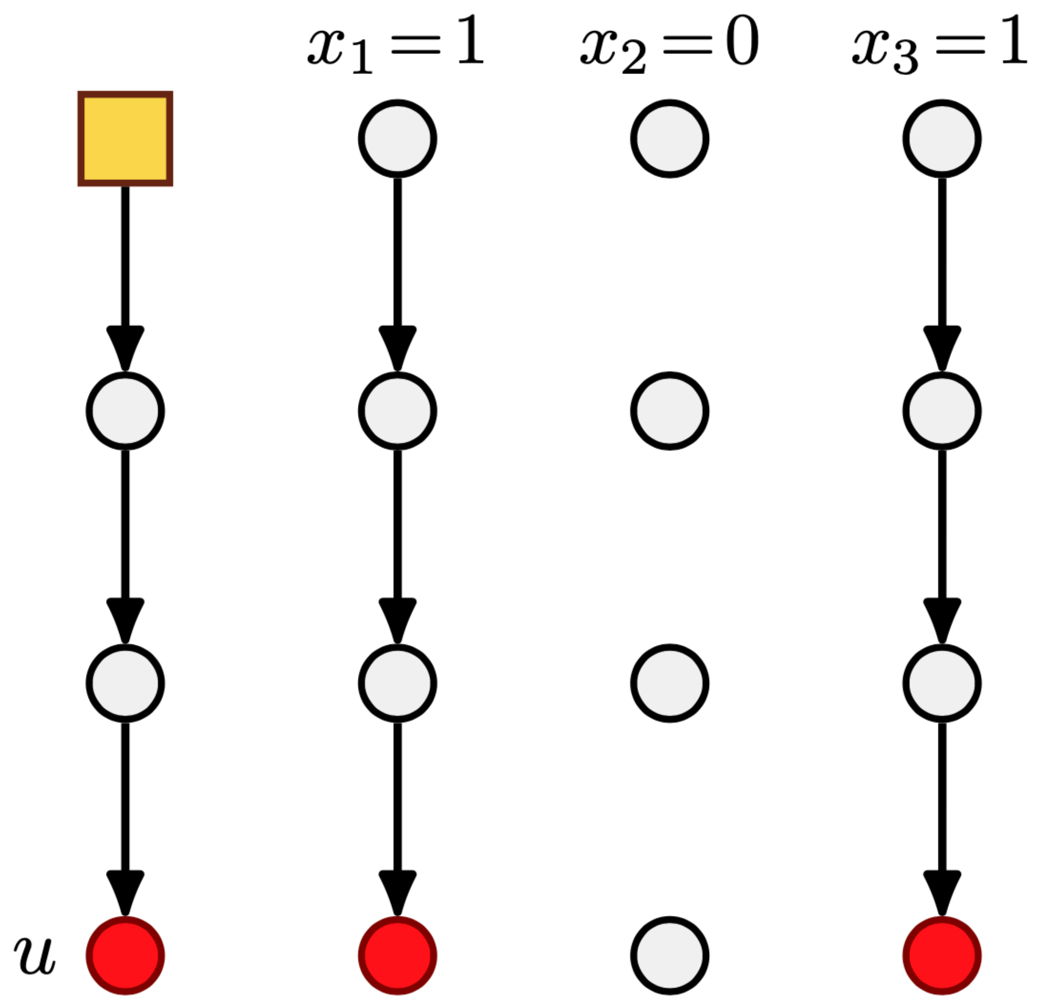
$$\deg_{\varepsilon\text{-NS}}(SoPL_n) \leq \deg_{\varepsilon}(OR_n)$$

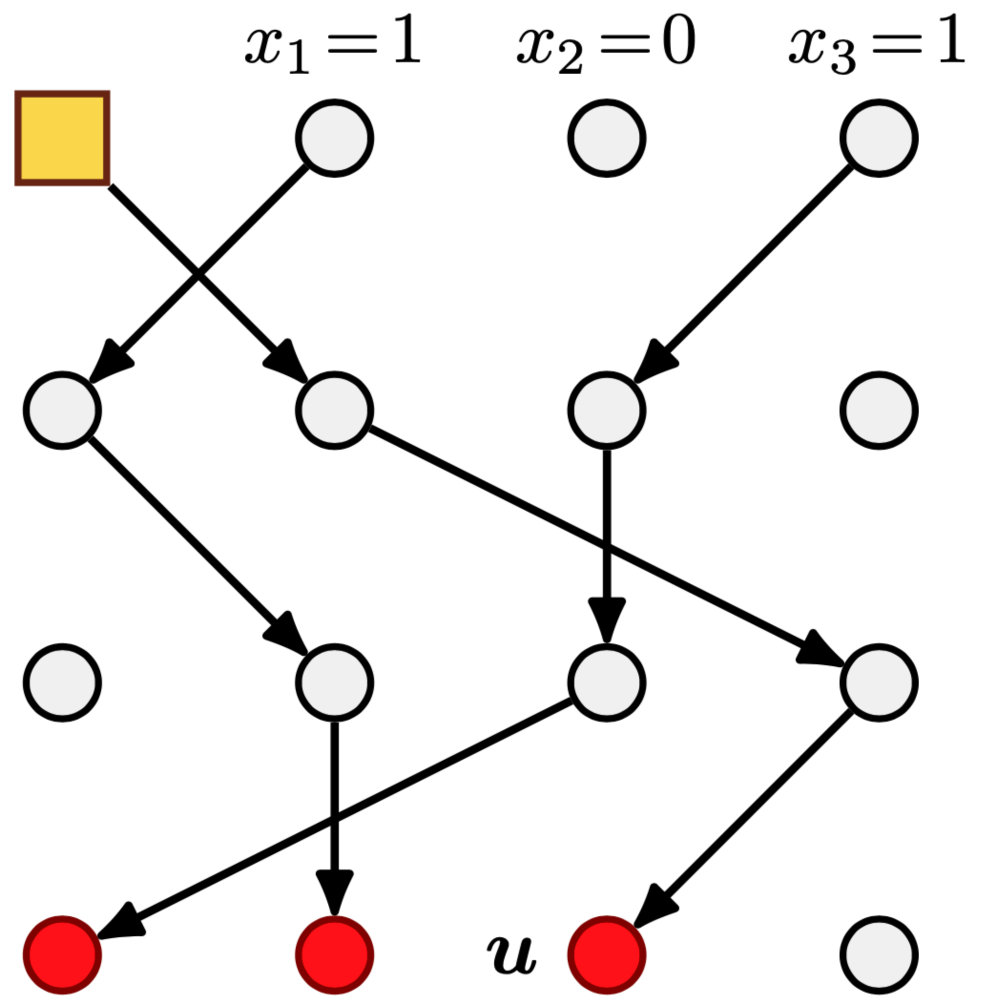
Lemma: Every $\frac{1}{2}$-NS refutation of SoPL_n requires deg $n^{\Omega(1)}$.

IDEA: Randomized decision-to-search reduction in the style of Raz-Wigderson '92 (matching)

We show ε-NS proofs for SoPL $\Rightarrow$ apx poly for OR

$$\deg_{\varepsilon\text{-NS}}(\mathrm{SoPL}_n) \leq \deg_{\varepsilon}(\mathrm{OR}_n)$$





What's a Reduction?

A pair (f, u) s.t.

What's a Reduction?

A pair (f, u) s.t.

(i) $f : \{0,1\}^{n-1} \to \{0,1\}^{O(n^2 \log n)}$ mapping inputs of

OR_{n-1} to $SoPL_n$

What's a Reduction?

A pair (f, u) s.t.

(i) $f : \{0,1\}^{n-1} \to \{0,1\}^{O(n^2 \log n)}$ mapping inputs of OR_{n-1} to $SoPL_n$ s.t. $f_i(x)$ is depth d decision tree

What's a Reduction?

A pair (f, u) s.t.

(i) $f : \{0,1\}^{n-1} \to \{0,1\}^{O(n^2 \log n)}$ mapping inputs of OR_{n-1} to $SoPL_n$ s.t. $f_i(x)$ is depth d decision tree

(ii) For any x, the only solutions of $y = f(x)$ are active sinks on the last row.

What's a Reduction?

A pair (f, u) s.t.

(i) $f : \{0,1\}^{n-1} \to \{0,1\}^{O(n^2 \log n)}$ mapping inputs of OR_{n-1} to $SoPL_n$ s.t. $f_i(x)$ is depth d decision tree

(ii) For any x, the only solutions of $y = f(x)$ are active sinks on the last row.

Moreover, u is a planted solution

What's a Reduction?

A pair (f, u) s.t.

(i) $f : \{0,1\}^{n-1} \to \{0,1\}^{O(n^2 \log n)}$ mapping inputs of OR_{n-1} to $SoPL_n$ s.t. $f_i(x)$ is depth d decision tree

(ii) For any x, the only solutions of $y = f(x)$ are active sinks on the last row.
Moreover, u is a planted solution

(iii) $OR(x) = \begin{cases} 0 \\ 1 \end{cases}$ then $\begin{array}{l} Sol(y) = \{u\} \\ |Sol(y)| \geq 2 \end{array}$

What's a Reduction?

A pair (f, u) s.t.

(i) $f : \{0,1\}^{n-1} \to \{0,1\}^{O(n^2 \log n)}$ mapping inputs of OR_{n-1} to $SoPL_n$ s.t. $f_i(x)$ is depth d decision tree

(ii) For any x, the only solutions of $y = f(x)$ are active sinks on the last row.

Moreover, u is a planted solution

(iii) $OR(x) = \begin{cases} 0 \\ 1 \end{cases}$ then $\begin{array}{l} Sol(y) = \{u\} \\ |Sol(y)| \geq 2 \end{array}$

↳ Randomised reduction is a distribution over reductions

Ideal Reduction $\not\Rightarrow$ Apx to OR

Ideal Reduction $\not\Rightarrow$ Apx to OR

↳ Ideal (y,u): for every y, $(u|y=y)$ is uniform over $Sol(y)$

Ideal Reduction ⇒ Apx to OR

↳ Ideal (y,u): for every y, $(u \mid \mathbf{y}=y)$ is uniform over $\mathrm{Sol}(y)$

$$\mathbb{E}[q(\mathbf{y},\mathbf{u})] = \mathbb{E}_{y\sim \mathbf{y}}\left[\mathbb{E}_{u'\sim(\mathbf{u}\mid \mathbf{y}=y)}\left[p_{i_{u'}}(y)\, a_{i_{u'}}(y)\right]\right]$$

Ideal Reduction ⇒ Apx to OR

↳ Ideal (y,u): for every y, $(u \mid y=y)$ is uniform over $Sol(y)$

$$\mathbb{E}[q(y,u)] = \mathbb{E}_{y\sim y}\Big[\mathbb{E}_{u'\sim(u \mid y=y)}\big[p_{i_{u'}}(y)\, a_{i_{u'}}(y)\big]\Big]$$

$$= \mathbb{E}_{y\sim y}\Big[\mathbb{E}_{u'\sim Sol(y)}\big[p_{i_{u'}}(y)\, a_{i_{u'}}(y)\big]\Big]$$

cause it's IDEAL

Ideal Reduction ⇒ Apx to OR

↳ Ideal (y,u): for every y, $(u \mid y = y)$ is uniform over $Sol(y)$

$$\mathbb{E}[q(y,u)] = \mathop{\mathbb{E}}_{y \sim y}\left[\mathop{\mathbb{E}}_{u' \sim (u \mid y = y)}\left[p_{i_{u'}}(y)\, a_{i_{u'}}(y)\right]\right]$$

$$= \mathop{\mathbb{E}}_{y \sim y}\left[\mathop{\mathbb{E}}_{u' \sim Sol(y)}\left[p_{i_{u'}}(y)\, a_{i_{u'}}(y)\right]\right]$$

cause it's IDEAL

$$= \mathop{\mathbb{E}}_{y \sim y}\left[\frac{1}{|Sol(y)|} \sum_{u'} p_{i_{u'}}(y)\, a_{i_{u'}}(y)\right]$$

Ideal Reduction $\Rightarrow$ Apx to OR

↳ Ideal (y,u): for every y, $(u \mid \mathbf{y}=y)$ is uniform over $Sol(y)$

$$\mathbb{E}[q(\mathbf{y},\mathbf{u})] = \mathbb{E}_{y\sim\mathbf{y}}\left[\mathbb{E}_{u'\sim(\mathbf{u}\mid\mathbf{y}=y)}\left[p_{i_{u'}}(y)\,a_{i_{u'}}(y)\right]\right]$$

cause it's IDEAL

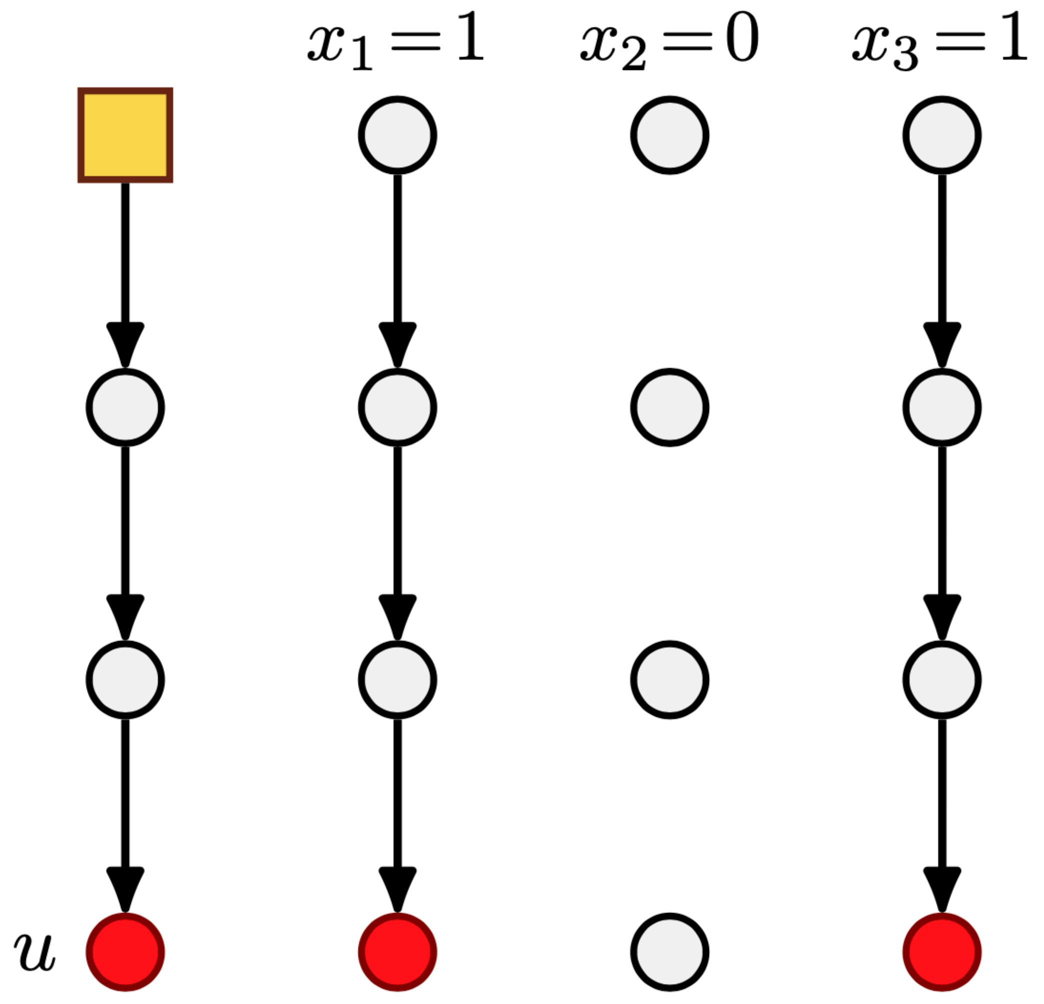
$$= \mathbb{E}_{y\sim\mathbf{y}}\left[\mathbb{E}_{u'\sim Sol(y)}\left[p_{i_{u'}}(y)\,a_{i_{u'}}(y)\right]\right]$$

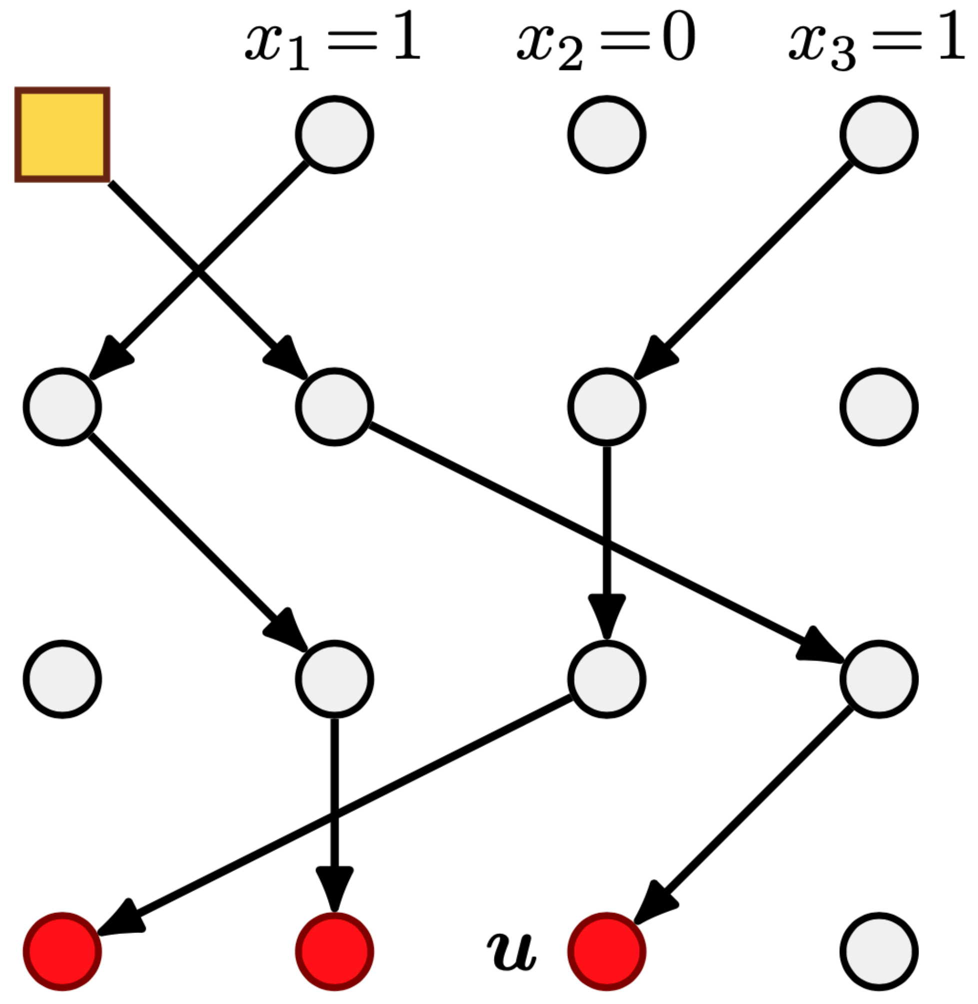
$$= \mathbb{E}_{y\sim\mathbf{y}}\left[\frac{1}{|Sol(y)|}\sum_{u'} p_{i_{u'}}(y)\,a_{i_{u'}}(y)\right]$$

$$= \frac{1\pm\varepsilon}{|Sol(y)|}$$

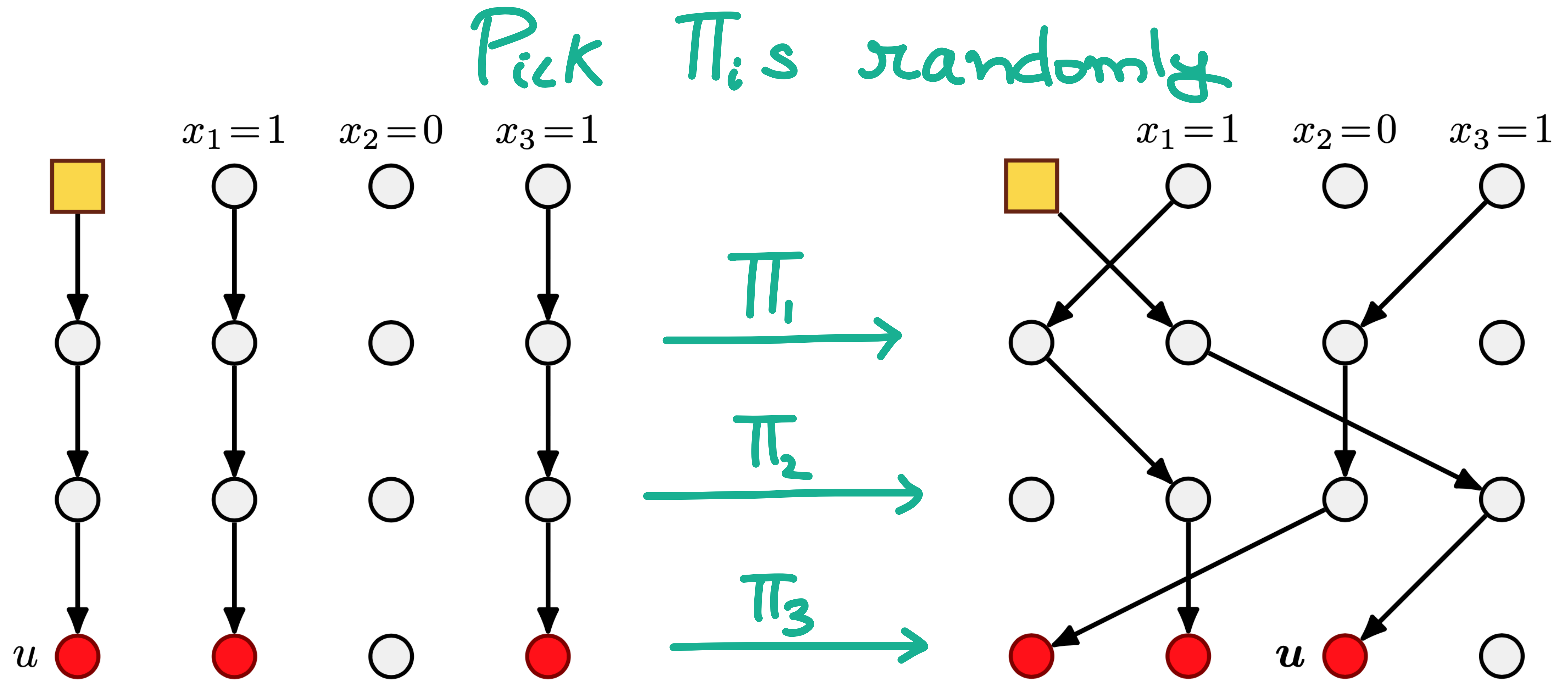
using that it's an ε-NS proof

Our Reduction



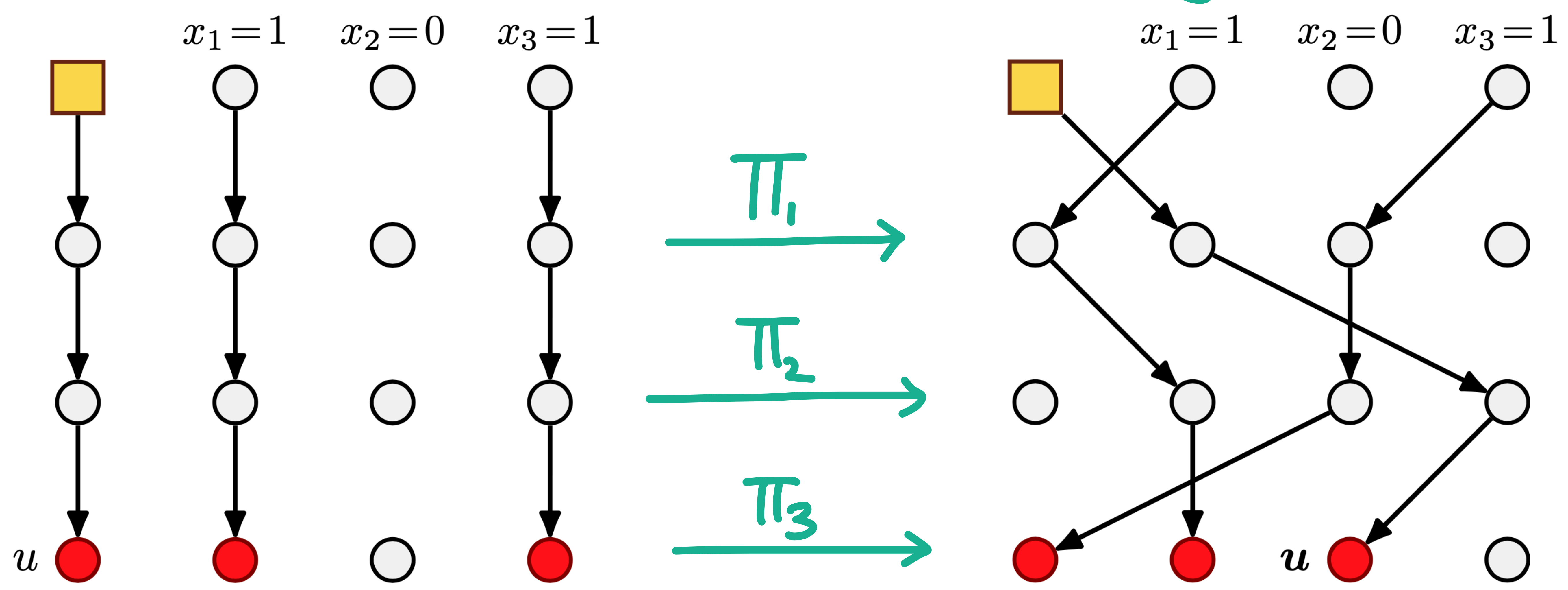


Our Reduction



Our Reduction

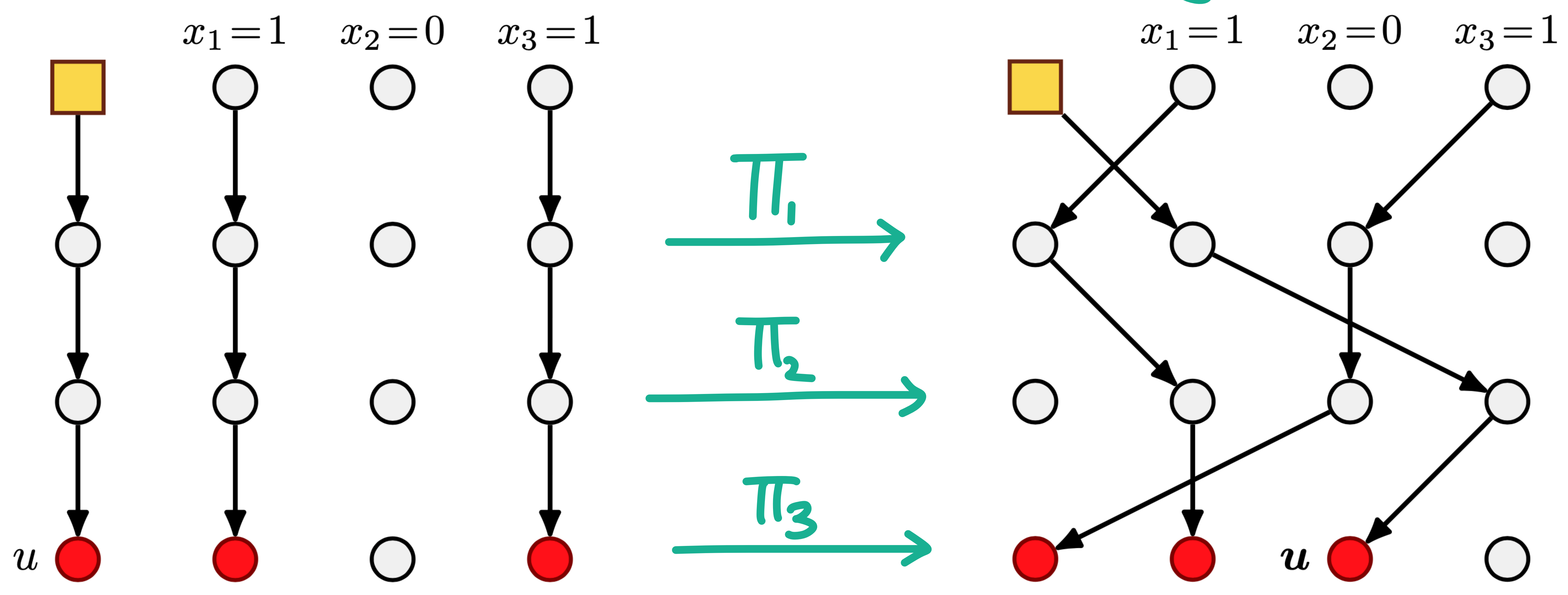
Pick π_is randomly



Is this Ideal?

Our Reduction

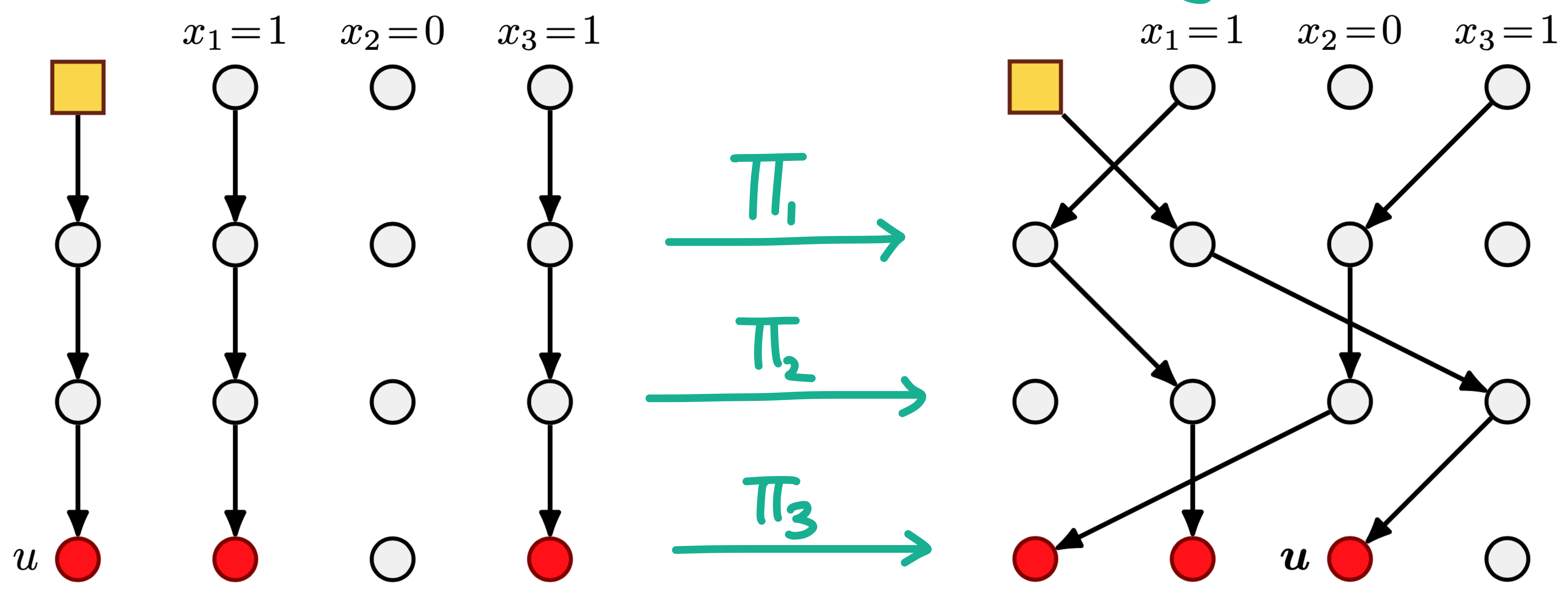
Pick π_is randomly



Is this Ideal? NO.

Our Reduction

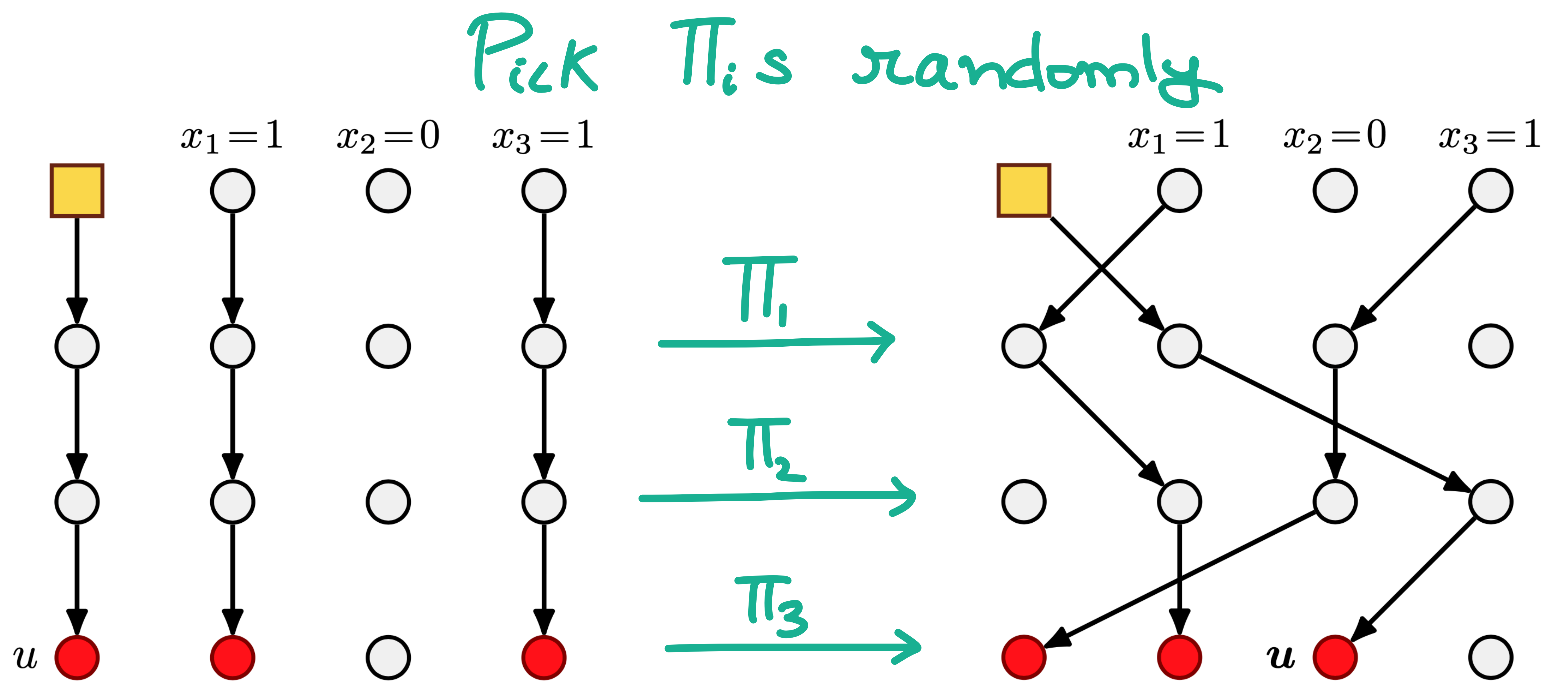
Pick π_is randomly



Is this Ideal? NO.

Claim This reduction is Locally Ideal, which suffices.

Our Reduction



Is this Ideal? NO.

Claim This reduction is Locally Ideal, which suffices.

Note $|Sol(y)| = 1 + |x|$ w.p. 1

$$r(x) = \mathbb{E}_{R_x}[q(y,u)]$$ R_x: "Our Reduction"

$$r(x) = \mathbb{E}_{R_x}[q(y,u)] \qquad R_x : \text{“Our Reduction”}$$

$$r'(x) = \mathbb{E}_{I_x}[q(y,u)] \qquad I_x : \text{An Ideal Reduction}$$

$$r(x) = \mathbb{E}_{R_x}[q(y,u)]$$ R_x : "Our Reduction"

$$r'(x) = \mathbb{E}_{I_x}[q(y,u)]$$ I_x : An Ideal Reduction

Claim $r(x) = r'(x)$ for all $x \in \{0,1\}^{n-1}$

$$r(x) = \mathbb{E}_{R_x}[q(y,u)]$$ R_x : "Our Reduction"

$$r'(x) = \mathbb{E}_{I_x}[q(y,u)]$$ I_x : An Ideal Reduction

Claim $r(x) = r'(x)$ for all $x \in \{0,1\}^{n-1}$

Proof By Linearity of Expectation, enough to show $\mathbb{E}_{R_x}[m(y,u)] = \mathbb{E}_{I_x}[m(y,u)]$ for any monomial.

$r(x) = \mathbb{E}_{R_x}[q(y,u)]$ R_x : "Our Reduction"

$r'(x) = \mathbb{E}_{I_x}[q(y,u)]$ I_x : An Ideal Reduction

Claim $r(x) = r'(x)$ for all $x \in \{0,1\}^{n-1}$

Proof By Linearity of Expectation, enough to show

$\mathbb{E}_{R_x}[m(y,u)] = \mathbb{E}_{I_x}[m(y,u)]$ for any monomial.

We assume $\deg(m) = o(\mathrm{poly}(n))$ ← else nothing to prove

$r(x) = \mathbb{E}_{R_x}[q(y,u)]$ R_x: "Our Reduction"

$r'(x) = \mathbb{E}_{I_x}[q(y,u)]$ I_x: An Ideal Reduction

Claim $r(x) = r'(x)$ for all $x \in \{0,1\}^{n-1}$

Proof By Linearity of Expectation, enough to show

$\mathbb{E}_{R_x}[m(y,u)] = \mathbb{E}_{I_x}[m(y,u)]$ for any monomial.

We assume $\deg(m) = o(\text{poly}(n))$ ← else nothing to prove

$\Rightarrow \exists\ i \in [\frac{n}{3}, \frac{2n}{3}]$ s.t. m reads none of the vars in rows $i, i+1$

$r(x) = \mathbb{E}_{R_x}[q(y,u)]$ R_x: "Our Reduction"

$r'(x) = \mathbb{E}_{I_x}[q(y,u)]$ I_x: An Ideal Reduction

Claim $r(x) = r'(x)$ for all $x \in \{0,1\}^{n-1}$

Proof By Linearity of Expectation, enough to show

$\mathbb{E}_{R_x}[m(y,u)] = \mathbb{E}_{I_x}[m(y,u)]$ for any monomial.

We assume $\deg(m) = o(\mathrm{poly}(n))$ ← else nothing to prove

$\Rightarrow \exists\, i \in [\frac{n}{3}, \frac{2n}{3}]$ s.t. m reads none of the vars in rows $i, i+1$

$\Rightarrow$ Given $(y,u) \sim R_x$ let $A \subseteq \{i\} \times [n+1]$, $B \subseteq \{i+1\} \times [n+1]$ be the active nodes. We can apply a random bijection $A \to B$ and get an Ideal reduction.

On Separations

Lemma: Every $\frac{1}{2}$-NS refutation of $SOPL_n$ requires deg $n^{\Omega(1)}$.

On Separations

Lemma: Every $\frac{1}{2}$-NS refutation of $SOPL_n$ requires deg $n^{\Omega(1)}$.

How do we amplify coefficients? Repeat instance!

On Separations

Lemma: Every $\frac{1}{2}$-NS refutation of $SOPL_n$ requires deg $n^{\Omega(1)}$.

How do we amplify coefficients? Repeat instance!

Lemma: Any degree $n^{o(1)}$ SA proof of SoD_{n^2} requires coefficients of magnitude $\exp(\Omega(n))$.

On Separations

Lemma: Every $\frac{1}{2}$-NS refutation of $SOPL_n$ requires deg $n^{\Omega(1)}$.

How do we amplify coefficients? Repeat instance!

Lemma: Any degree $n^{o(1)}$ SA proof of SoD_{n^2} requires coefficients of magnitude $\exp(\Omega(n))$.

Hard instance for ε-NS $\rightarrow$ l.b. on $J(x)$ in SA

$$\sum_i p_i(x)\,a_i(x) = 1 + J(x)$$

$$\sum_i \underbrace{p_i(x)\,a_i(x)}_{\deg < n^{o(1)}} = \underbrace{1 + J(x)}_{\geq 2^{\Omega(n)}}$$

$$\sum_i \underbrace{p_i(x)a_i(x)}_{\deg < n^{o(1)}} = \underbrace{1 + J(x)}_{\geq 2^{\Omega(n)}}$$

$\Rightarrow$ Some monomial with coefficient $\geq \dfrac{2^{\Omega(n)}}{n^{n^{o(1)}}} = 2^{\mathrm{poly}(n)}$

$$\sum_i \underbrace{p_i(x)\,a_i(x)}_{\deg < n^{o(1)}} = \underbrace{1 + J(x)}_{\geq 2^{\Omega(n)}}$$

$\Rightarrow$ Some monomial with coefficient $\geq \dfrac{2^{\Omega(n)}}{n^{n^{o(1)}}} = 2^{\mathrm{poly}(n)}$

Thanks for your attention!