Derivation reduction of metarules in meta-interpretive learning

Andrew Cropper & Sophie Tourret

Input	**Output**
Examples Background knowledge **Bias**	Logic program

Biases

- Mode declarations (Progol, ILASP, Aleph, XHAIL, …)
- **Metarules** (Metagol, MIL-Hex, ∂ILP, ProPPR, Clint, MOBAL …)

Metarules

∃PQ∀AB	P(A,B) ← Q(A,B)
∃PQR∀AB	P(A,B) ← Q(A),R(A,B)
∃PQR∀ABC	P(A,B) ← Q(A,C),R(C,B)

Metarules

```
P(A,B) ← Q(A,B)
P(A,B) ← Q(A),R(A,B)
P(A,B) ← Q(A,C),R(C,B)
```

P,*Q*,*R* are **existentially** quantified **second-order** variables

A,B,*C* are **universally** quantified **first-order** variables

Input	Output
% background parent(ann,amy)← parent(amy,amelia)← % example grandparent(ann,amelia)← % metarule P(A,B)←Q(A,C),R(C,B)	

Input	Output

% background
parent(ann,amy)←
parent(amy,amelia)←
 	
% example
grandparent(ann,amelia)←
 	
% metarule
P(A,B)←Q(A,C),R(C,B)
	grandparent(A,B)←
	    parent(A,C),
	    parent(C,B)
	
	{
	 P\granparent,
	 Q\parent,
	 R\parent
	}

Why?

Completeness

cannot learn grandparent/2 with only `P(X)←Q(X)`

Efficiency

more metarules = larger hypothesis space

Usability

Users do not want to provide metarules

Remove redundant metarules [ILP14]

The Horn clause C is **entailment redundant** in the Horn theory $T \cup \{C\}$ when $T \models C$

Entailment redundancy

```
C1 = h(A,B)←s(A,B)
C2 = h(A,B)←s(A,B),u(B)
C3 = h(A,B)←s(A,B),u(A,B)
C4 = h(A,B)←s(A,B),u(A,B),v(A,B)
```

Entailment redundancy

```
C1 = h(A,B)←s(A,B)
```

~~C2 = h(A,B)←s(A,B),u(B)~~

~~C3 = h(A,B)←s(A,B),u(A,B)~~

~~C4 = h(A,B)←s(A,B),u(A,B),v(A,B)~~

{C1} ⊨ {C2,C3,C4}

Entailment reduction of metarules [ILP14]

P(A,B)←Q(A,B)
P(A,B)←Q(B,A)
P(A,B)←Q(A,C),R(B,C)
P(A,B)←Q(A,C),R(C,B)
P(A,B)←Q(B,A),R(A,B)
P(A,B)←Q(B,A),R(B,A)
P(A,B)←Q(B,C),R(A,C)
P(A,B)←Q(B,C),R(C,A)
P(A,B)←Q(C,A),R(B,C)
P(A,B)←Q(C,A),R(C,B)
P(A,B)←Q(C,B),R(A,C)
P(A,B)←Q(C,B),R(C,A)

→ ?

Entailment reduction of metarules [ILP14]

P(A,B)←Q(A,B)
P(A,B)←Q(B,A)
P(A,B)←Q(A,C),R(B,C)
P(A,B)←Q(A,C),R(C,B)
P(A,B)←Q(B,A),R(A,B)
P(A,B)←Q(B,A),R(B,A)
P(A,B)←Q(B,C),R(A,C)
P(A,B)←Q(B,C),R(C,A)
P(A,B)←Q(C,A),R(B,C)
P(A,B)←Q(C,A),R(C,B)
P(A,B)←Q(C,B),R(A,C)
P(A,B)←Q(C,B),R(C,A)

→

P(A,B)←Q(B,A)

P(A,B)←Q(A,C),R(C,B)

Entailment redundancy

```
C1 = P(A,B)←Q(A,B)
C2 = P(A,B)←Q(A,B),R(A)
C3 = P(A,B)←Q(A,B),R(A,B)
C4 = P(A,B)←Q(A,B),R(A,B),S(A,B)
```

Entailment redundancy

```
C1 = P(A,B)←Q(A,B)
```

~~C2 = P(A,B)←Q(A,B),R(A)~~

~~C3 = P(A,B)←Q(A,B),R(A,B)~~

~~C4 = P(A,B)←Q(A,B),R(A,B),S(A,B)~~

$\{C1\} \models \{C2, C3, C4\}$

Entailment redundancy

```
C1 = P(A,B)←Q(A,B)
```

~~C2 = P(A,B)←Q(A,B),R(A)~~

~~C3 = P(A,B)←Q(A,B),R(A,B)~~

~~C4 = P(A,B)←Q(A,B),R(A,B),S(A,B)~~

{C1} ⊨ {C2,C3,C4}

father(A,B)←parent(A,B),male(A) ✖

Derivation redundancy

The Horn clause C is **derivationally redundant** in the Horn theory $T \cup \{C\}$ when $T \vdash C$

↑ SLD-resolution

Derivation redundancy

```
C1 = P(A,B)←Q(A,B)
C2 = P(A,B)←Q(A,B),R(A)
C3 = P(A,B)←Q(A,B),R(A,B)
C4 = P(A,B)←Q(A,B),R(A,B),S(A,B)
```

Derivation redundancy

```
C1 = P(A,B)←Q(A,B)
C2 = P(A,B)←Q(A,B),R(A)
C3 = P(A,B)←Q(A,B),R(A,B)
```

~~C4 = P(A,B)←Q(A,B),R(A,B),S(A,B)~~

$\{C1,C2,C3\} \vdash \{C4\}$

father(A,B)←parent(A,B),male(A) ✓

Derivation redundancy

While there is a clause in T such that T - {C} $\vdash_k$ C:

Set T = T - {C}

Connected clauses

body literals are connected to the head literal

```
P(A)←Q(A) ✓
P(A,B)←Q(A,C) ✓
P(A,B)←Q(A,B),R(B,D),S(D,B) ✓

P(A)←Q(B) ✖
P(A)←Q(A),R(B,C) ✖
P(A,B)←Q(A,B),S(C) ✖
```

H^2_m

restriction on literal arity

```
P(A,B)←Q(A,B) ✓
P(A)←Q(A,B),R(B) ✓

P(A,B,C)←Q(A,B,C) ✖
P(A)←Q(A,B,C),R(B,C) ✖
```

$H^{2=}_{m}$

P(A,B)←Q(A,B) ✓
P(A,B)←Q(A,C),R(C,B) ✓

P(A)←**Q(A)** ✖
P(A,B)←Q(A,B),**R(B)** ✖

H^a_2

restriction on number of body literals

P(A,B)←Q(A,B) ✓
P(A)←Q(A,B,C),R(B,C) ✓

P(A)←**Q(A),R(A),S(A)** ✖
P(A,B)←**Q(A),R(B),S(A,B)** ✖

$H^{a}_{2=}$

P(A)←Q(A),R(A) ✓

P(A,B)←Q(A,B),R(A,B) ✓

P(A)←**Q(A)** ✖

P(A,B)←**Q(A,B),R(B)** ✖

Exactly-two connected

each first-order variable appears exactly twice

P(A)←Q(A) ✓
P(A,B)←Q(A,B) ✓
P(A,B)←Q(A,C),R(C,B) ✓

P(A,**B**)←Q(A) ✖
P(A)←Q(A,**B**) ✖
P(**A**)←Q(**A**),R(**A**) ✖

Idea

1. Run **derivation reduction** with a SLD-resolution depth bound of **10** on **sub-fragments** of an **infinite** fragment.
2. Study the results.

$E^{2=}{}_{5}$

$E^{2=}_{5}$

E-reduction	D-reduction
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)

$E^{2=}_{5}$

E-reduction	D-reduction
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)

Same as ILP14 paper

$$E^{2=}_{2} \vdash E^{2=}_{\infty}$$ ✔

$E^2{}_5$

E^2_5

E-reduction	D-reduction
P(A)←Q(A)	P(A)←Q(A)
P(A)←Q(A,B),R(B)	P(A)←Q(A,B),R(B)
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A),R(B)	P(A,B)←Q(A),R(B)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)

$E^2{}_5$

E-reduction	D-reduction
P(A)←Q(A)	P(A)←Q(A)
P(A)←Q(A,B),R(B)	P(A)←Q(A,B),R(B)
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A),R(B)	P(A,B)←Q(A),R(B)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)

$$E^2{}_2 \vdash E^2{}_\infty$$ ✔

Two connected

each first-order variable appears at least twice
(i.e. prevents singleton variables)

P(A)←Q(A) ✓
P(A)←Q(A),R(A) ✓
P(A,B)←Q(A,B),R(B) ✓
P(A,B)←Q(A,C),R(C,B) ✓

P(A,**B**)←Q(A) ✖
P(A)←Q(A,**B**) ✖
P(A)←Q(A),R(A,**B**) ✖

$K^{2=}_5$ two connected

$K^{2=}_5$ two connected

E-reduction	D-reduction
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)
	P(A,B)←Q(A,B),R(A,B)
	P(A,B)←Q(A,B),R(A,C),S(C,D),T(C,D)
	P(A,B)←Q(A,C),R(A,C),S(B,D),T(B,D)
	P(A,B)←Q(A,C),R(A,D),S(B,C),T(B,D),U(C,D)

$K^{2=}_5$ two connected

E-reduction	D-reduction
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)
	P(A,B)←Q(A,B),R(A,B)
	P(A,B)←Q(A,B),R(A,C),S(C,D),T(C,D)
	P(A,B)←Q(A,C),R(A,C),S(B,D),T(B,D)
	P(A,B)←Q(A,C),R(A,D),S(B,C),T(B,D),U(C,D)

$K^{2=}_2 \nvdash K^{2=}_\infty$ ✖

K^2_5 two connected

K^2_5 two connected

E-reduction	D-reduction
P(A)←Q(A)	P(A)←Q(A)
P(A)←R(A,B),Q(A,B)	P(A)←R(A,B),Q(A,B)
	P(A)←Q(A),R(A)
	P(A)←Q(B),R(A,B)
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A),R(B)	P(A,B)←Q(A),R(B)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)
	P(A,B)←Q(A,B),R(A,B)
	P(A,B)←Q(A),R(A,B)
	P(A,B)←Q(A,C),R(A,D),S(C,B),T(B,D),U(C,D)

K^2_5 two connected

E-reduction	D-reduction
P(A)←Q(A)	P(A)←Q(A)
P(A)←R(A,B),Q(A,B)	P(A)←R(A,B),Q(A,B)
	P(A)←Q(A),R(A)
	P(A)←Q(B),R(A,B)
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A),R(B)	P(A,B)←Q(A),R(B)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)
	P(A,B)←Q(A,B),R(A,B)
	P(A,B)←Q(A),R(A,B)
	P(A,B)←Q(A,C),R(A,D),S(C,B),T(B,D),U(C,D)

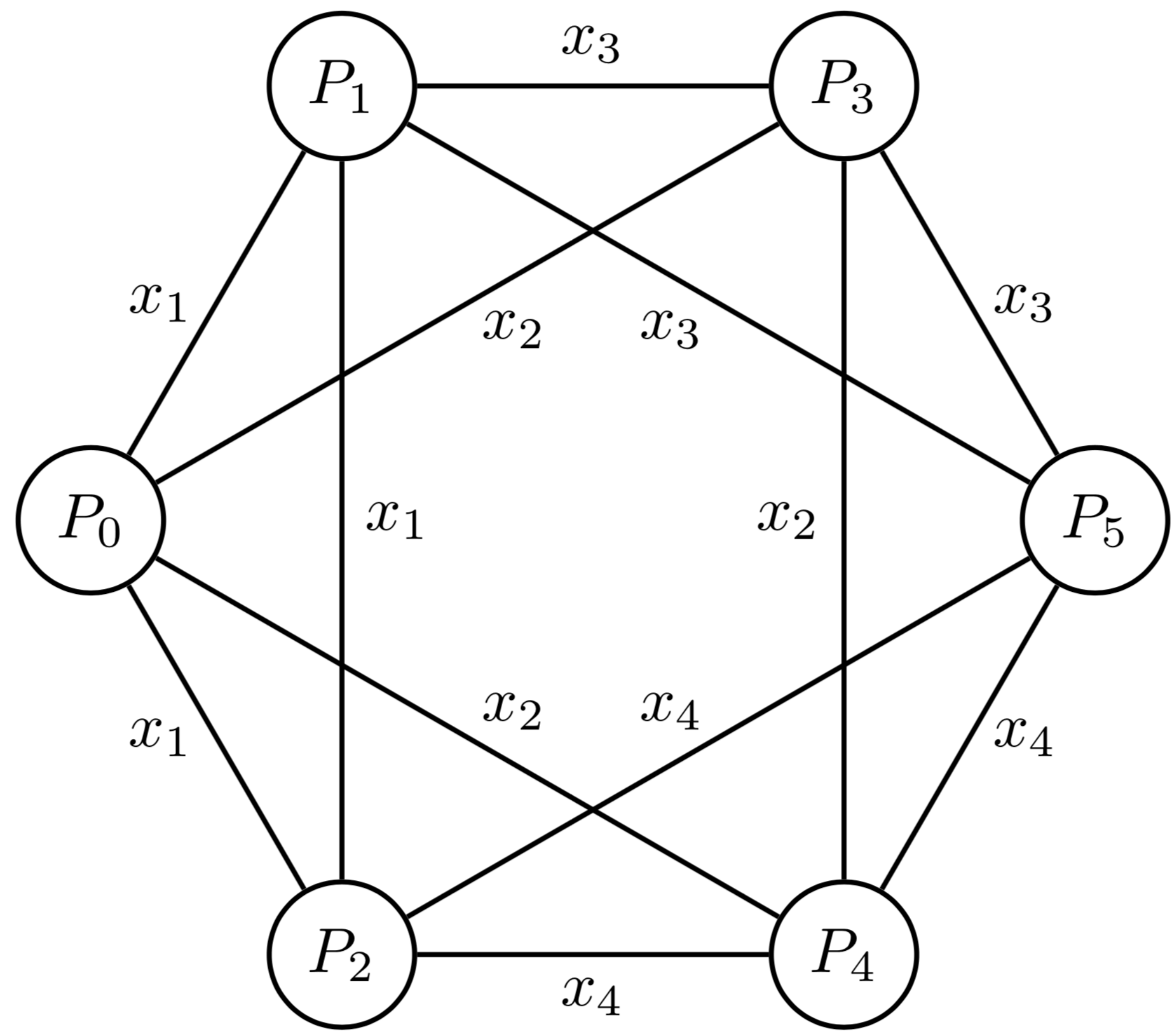
$$K^{2=}_2 \nvdash K^{2=}_5$$ ✖

K^2_5 two connected

E-reduction	D-reduction
P(A)←Q(A)	P(A)←Q(A)
P(A)←R(A,B),Q(A,B)	P(A)←R(A,B),Q(A,B)
	P(A)←Q(A),R(A)
	P(A)←Q(B),R(A,B)
P(A,B)←Q(B,A)	P(A,B)←Q(B,A)
P(A,B)←Q(A),R(B)	P(A,B)←Q(A),R(B)
P(A,B)←Q(A,C),R(C,B)	P(A,B)←Q(A,C),R(C,B)
	P(A,B)←Q(A,B),R(A,B)
	P(A,B)←Q(A),R(A,B)
	P(A,B)←Q(A,C),R(A,D),S(C,B),T(B,D),U(C,D)

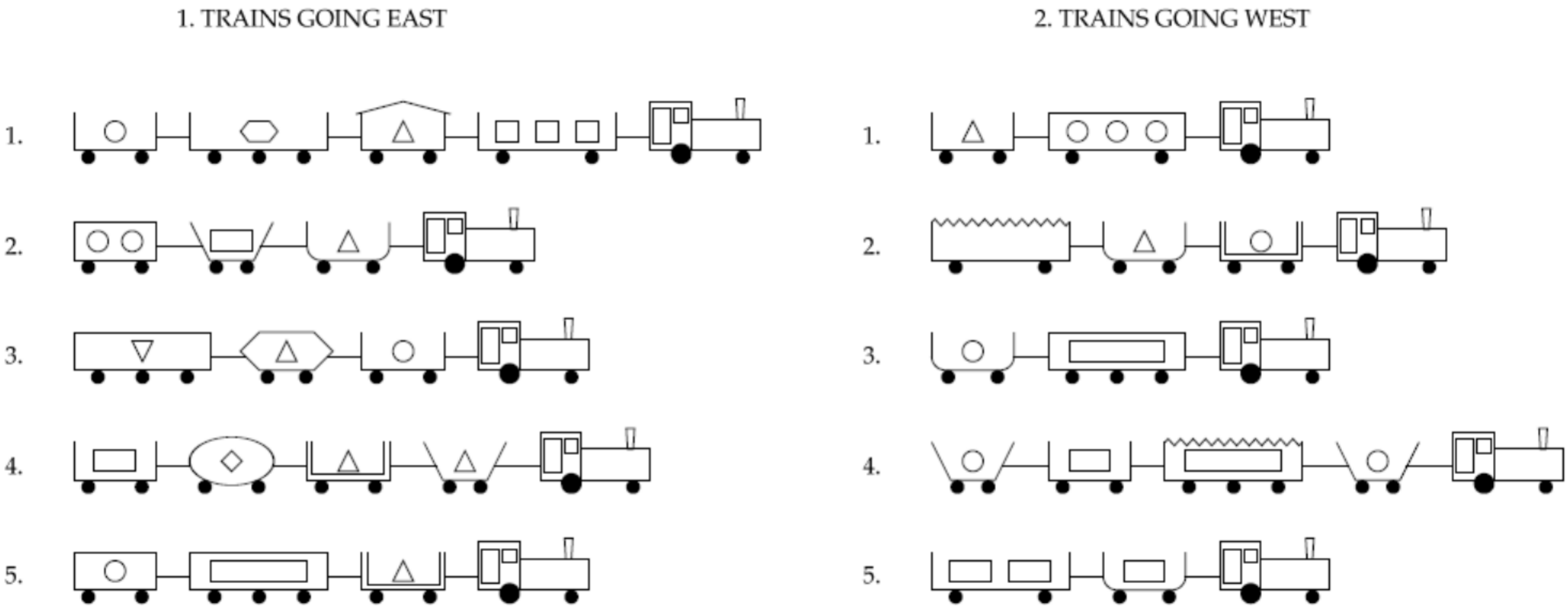
$K^{2=}_{\infty}$ cannot be reduced ✖

Why not?



$$P_0(x_1, x_2) \leftarrow P_1(x_1, x_3), P_2(x_1, x_4), P_3(x_2, x_3), P_4(x_2, x_4), P_5(x_3, x_4)$$

Does it matter?



Accuracies

Task	E-reduction	D-reduction	D*-reduction
T1	95 ± 1	100 ± 0	100 ± 0
T2	99 ± 1	100 ± 0	100 ± 0
T3	56 ± 3	96 ± 2	96 ± 2
T4	69 ± 4	96 ± 2	96 ± 2
T5	59 ± 3	93 ± 3	93 ± 3
T6	50 ± 1	96 ± 3	96 ± 3
T7	68 ± 4	95 ± 2	95 ± 2
T8	54 ± 3	60 ± 3	90 ± 3

Learning times

Task	E-reduction	D-reduction	D*-reduction
T1	0.01 ± 0	0 ± 0	0 ± 0
T2	0.01 ± 0	0 ± 0	0 ± 0
T3	431 ± 59	0.01 ± 0	0.01 ± 0
T4	300 ± 68	0 ± 0	0.01 ± 0
T5	427 ± 60	1 ± 0.3	1 ± 0.41
T6	600 ± 0	1 ± 0.41	1 ± 0.42
T7	917 ± 535	1 ± 0.27	1 ± 0.36
T8	487 ± 51	360 ± 67	26 ± 5

```
% target program
f(X):-has_car(X,C1),
    long(C1),
    two_wheels(C1),
    has_car(X,C2),
    long(C2),
    three_wheels(C2).
```

```
% E-reduction
f(A):-has_car(A,B),f1(A,B).
f1(A,B):-has_car(A,C),f2(C,B).
f2(A,B):-long(A),three_wheels(B).
```

```
% D-reduction
f(A):-f1(A),f2(A).
f1(A):-has_car(A,B),three_wheels(B).
f2(A):-has_car(A,B),roof_open(B).
```

```
% D*-reduction
f(A):-f1(A),f2(A).
f1(A):-has_car(A,B),three_wheels(B).
f2(A):-has_car(A,B),f3(B).
f3(A):-long(A),two_wheels(A).
```

```prolog
% target program
f(X):-
    has_car(X,C1),
    long(C1),
    two_wheels(C1),
    has_car(X,C2),
    long(C2),
    three_wheels(C2).
```

```prolog
% D*-reduction
f(A):-
    has_car(A,B),
    three_wheels(B),
    has_car(A,C),
    long(C),
    two_wheels(C).
```

Todo

- Study derivation reduction problem
- Other fragments
  - Triadics
  - Connected
- Unconstrained resolution