Well-Founded SHACL and the Modal μ-Calculus

A talk by: Anouk Michelle Oudshoorn

Work in collaboration with: Magdalena Ortiz & Mantas Šimkus

TU Wien

Recursive SHACL

- Shape constraint: $s \leftarrow \varphi$

$$\varphi ::= s \mid c \mid A \mid \top \mid \neg\varphi \mid \varphi \wedge \varphi \mid \geq_n E.\varphi \mid eq(E,r) \mid disj(E,r) \mid closed(R)$$

shape names — nomi-nals — concept names — regular expressions on role names

- Semantics: existance of a shape assignment S, a set of shape atoms $(s(a))$

  + specific rules based on the type of semantics: supported / well-founded / stable model/...

  no — some minimal models

  + $s \leftarrow \varphi \rightsquigarrow s^{I,S} = (\varphi)^{I,S}$

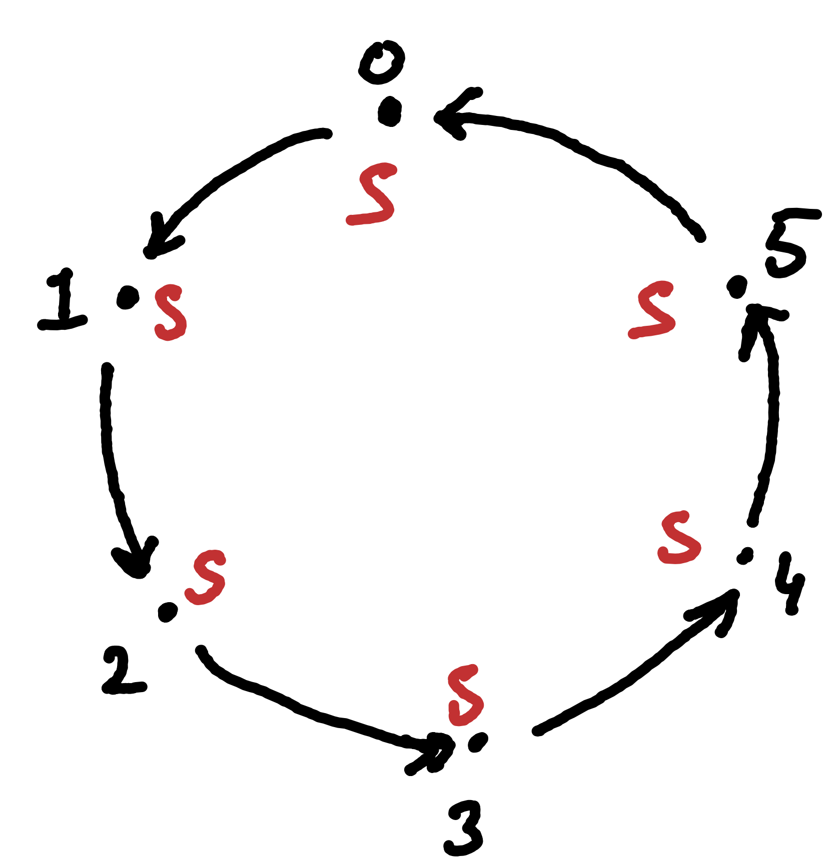
Example.

constraint:

$s \leftarrow A \vee \exists r.s$

target:

$s(0)$



$r^I := \{(0,1), \ldots, (4,5), (5,0)\}$

$A^I := \emptyset$

There is no well-founded assignment containing $s(0)$

'$\exists$' = '$\geq_1$'

Reasoning Problems

- Satisfiability (& finite sat)

  Exists $\mathcal{I}$ s.t. there is a shape assignment for $\mathcal{I}$ containing the targets.

- Containment/Implication (& finite cont./impl.)

  $(\mathcal{C}, \mathcal{G}) \subseteq (\mathcal{C}', \mathcal{G}')$ iff for all $\mathcal{I}$, if $\mathcal{I}$ validates $(\mathcal{C}, \mathcal{G})$, then $\mathcal{I}$ validates $(\mathcal{C}', \mathcal{G}')$.

And the following property:

- Finite Model Property

  Exists $\mathcal{I}$ that validates iff there exists a finite $\mathcal{I}$ that validates.

SHACL Satisfiability (for supported model semantics)

- Results can* be translated back & forth between $\mathcal{L}$ and $\mathcal{L}_S$.

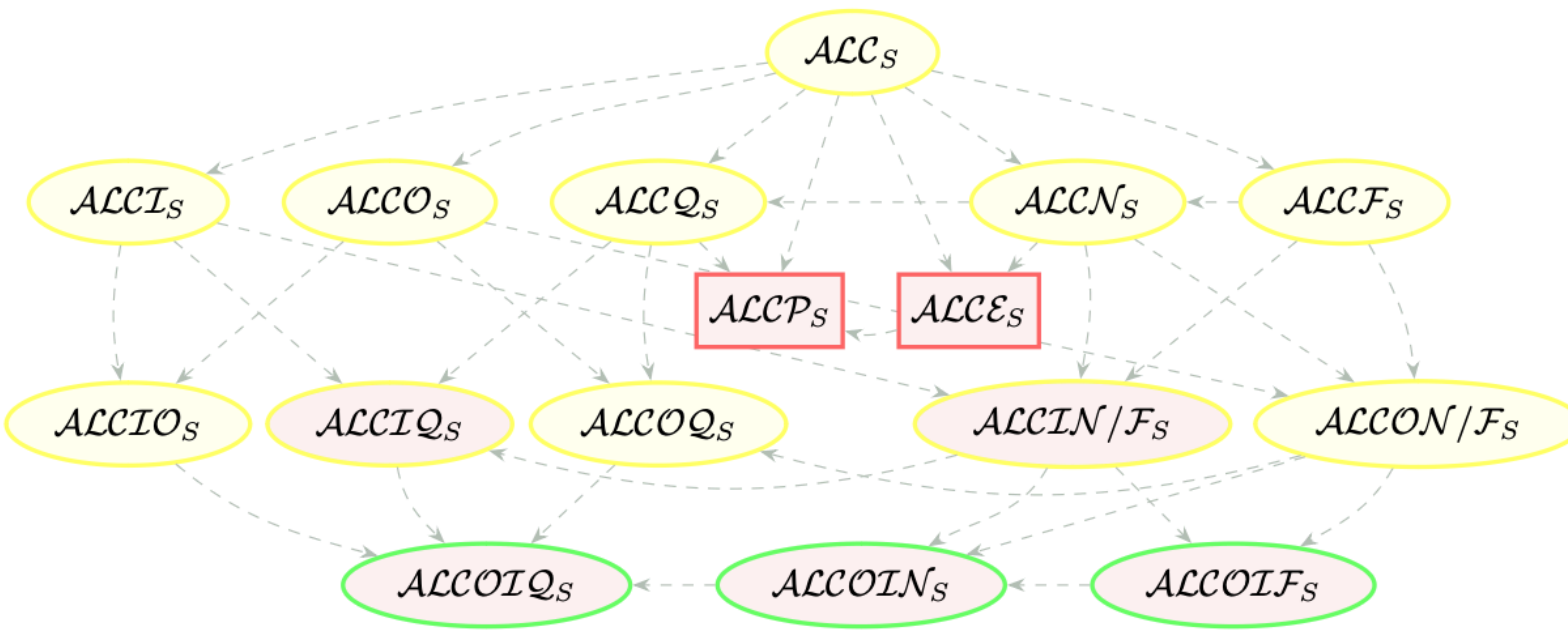


Figure 4: Decidability and complexity of SHACL fragments. Ellipse-shaped nodes denote (finite) satisfiability is decidable in ExpTime (yellow border), or NExpTime (green border). Squared-shaped nodes indicate satisfiability is undecidable. A yellow filling indicates the presence of the finite model property, whereas a red filling stands for the lack of it. Arrows indicate subsumption of fragments.

*: see [O25], Section 4 for details & exact conditions

Results from [O25] presented at DL'25.

SHACL Containment (for supported model semantics)

<u>Theorem</u> Deciding SHACL containment (supp. model sem.) is undecidable.

"$\mathcal{ALCOI}_S$"

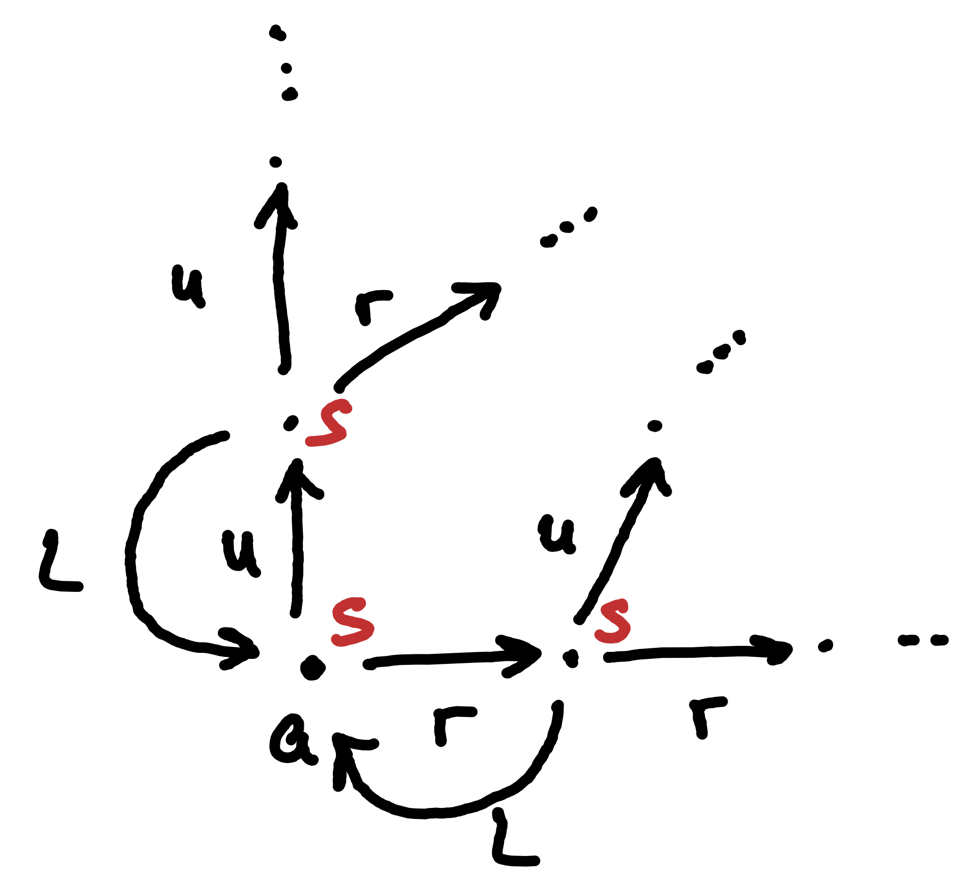
SHACL Containment (for supported model semantics)

Theorem Deciding SHACL containment (supp. model sem.) is undecidable.

"$\mathcal{ALCOI}_s$"

$\mathcal{C}: s \leftarrow \exists l.a \wedge \exists u.s \wedge \exists r.s$

$\mathcal{G}: s(a)$



$\mathcal{C}': s_a \leftarrow s_a$

$s_b \leftarrow s_b$

$s' \leftarrow \exists \bar{l}.(\exists r.\exists u.s_a \wedge \exists u.\exists r.s_b) \wedge$

$\forall \bar{l}.(s_a \vee s_b \wedge \neg(s_a \wedge s_b))$

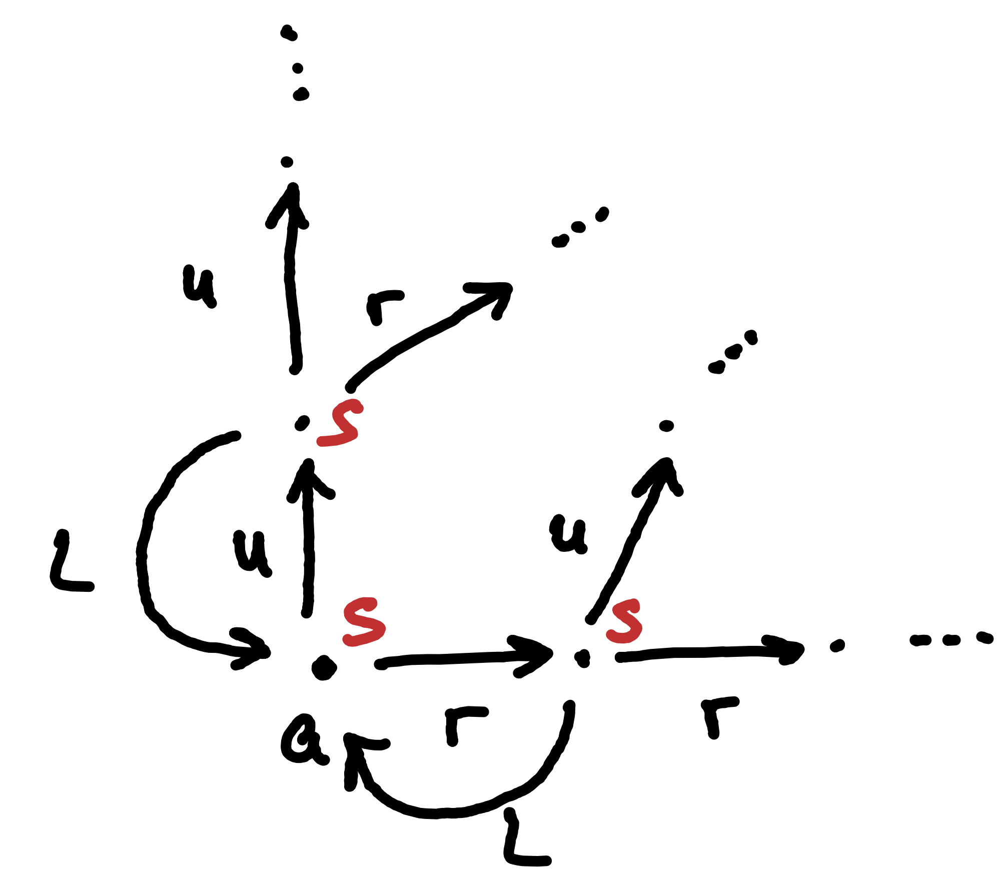
$\mathcal{G}': s'(a)$

SHACL Containment (for supported model semantics)

Theorem Deciding SHACL containment (supp. model sem.) is undecidable.
"$\mathcal{ALCOI}_s$"

$\mathcal{C}$: $s \leftarrow \exists l.a \wedge \exists u.s \wedge \exists r.s$

$\mathcal{G}$: $s(a)$



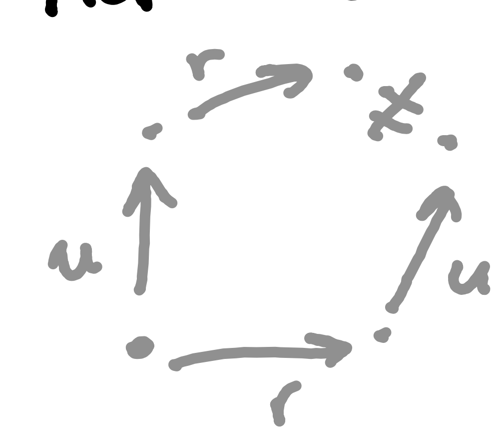
$\mathcal{C}'$: $s_a \leftarrow s_a$

$s_b \leftarrow s_b$

$s' \leftarrow \exists l^-.(\exists r.\exists u.s_a \wedge \exists u.\exists r.s_b) \wedge$
$\forall l^-.(s_a \vee s_b \wedge \neg(s_a \wedge s_b))$

$\mathcal{G}'$: $s'(a)$

Does there exist a structure validating $(\mathcal{C}, \mathcal{G})$ but not validating $(\mathcal{C}', \mathcal{G}')$?

$\exists l^-.(\exists r.\exists u.s_a \wedge \exists u.\exists r.s_b)$ – only possible when  exists somewhere

$\forall l^-.(s_a \vee s_b \wedge \neg(s_a \wedge s_b))$ – exclusive or, possible on each structure

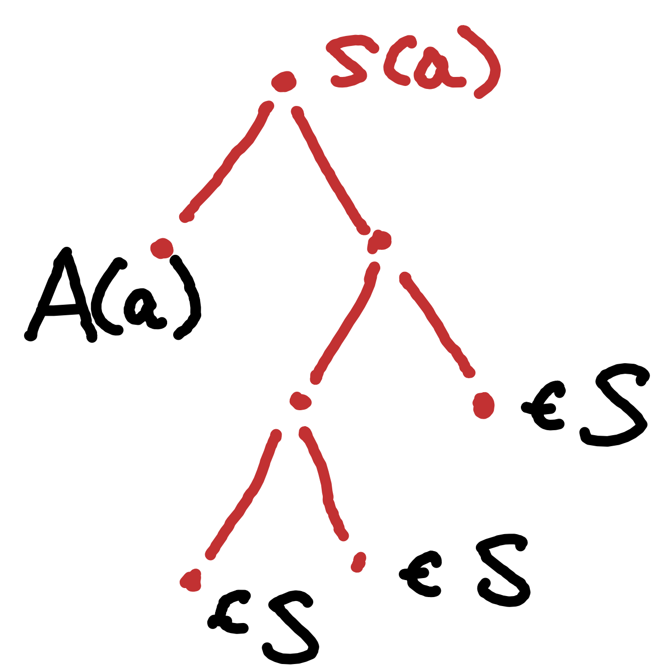
Adding tiling constraints to $(\mathcal{C}, \mathcal{G})$ finishes the argument.

Well-founded semantics

- Based on a least-fixed point operator:

$$W_{I,C}(S) = \underline{T_{I,C}(S)} \cup \neg . \underline{U_{I,C}(S)}$$

- In each round, finite, positive derivations & the greatest unfounded set are added.



$$\{\neg s(b) \;\; \cdots \;\; \neg s''(b) \;\; \neg s'(c) \;\; \cdots\} \cup S$$

given e, $\Downarrow$ does NOT derive

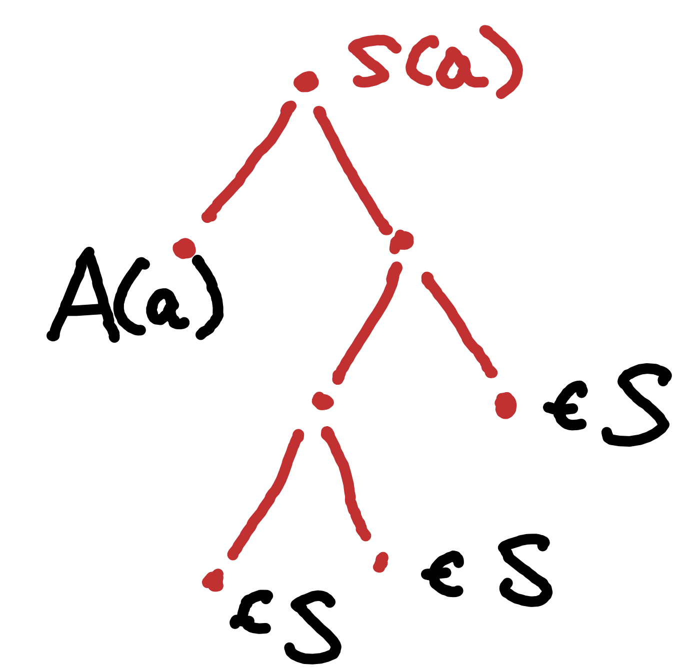
s(b) or s''(b) or s'(c) or ...

Well-founded semantics

- Based on a least-fixed point operator:

$$W_{I,C}(S) = \underline{T_{I,C}(S)} \cup \neg.\underline{U_{I,C}(S)}$$

- In each round, finite, positive derivations & the greatest unfounded set are added.

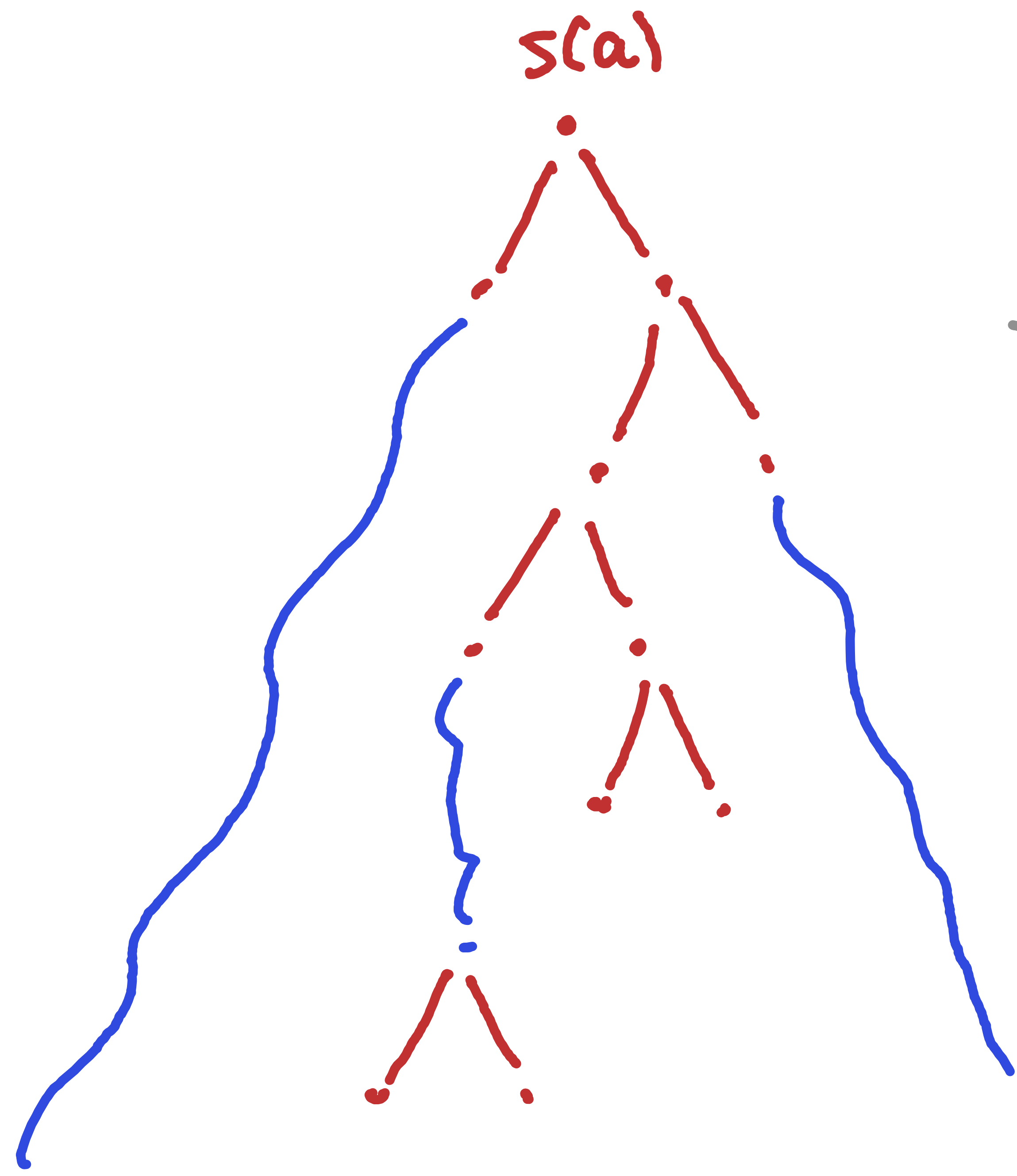


$$\{\neg s(b) \ \dots \ \neg s''(b) \quad \neg s'(c) \ \dots\} \cup S$$

given e, $\Downarrow$ does NOT derive

s(b) or s''(b) or s'(c) or ...

- An atom is well-founded if:

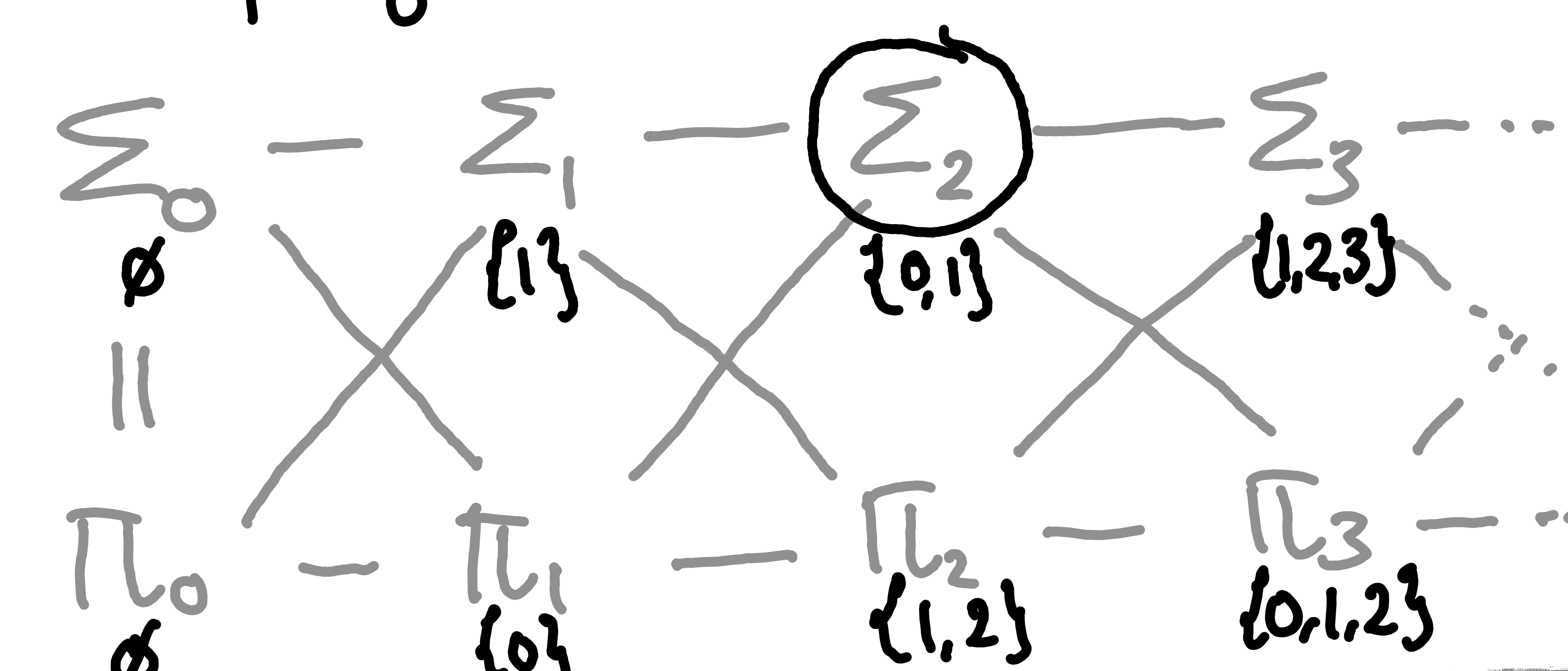


For each branch: $\exists n \in \mathbb{N}$ s.t. no red after n steps.

- Note the least- and greatest- fixed points already showing up...

$$\mu X \dots \nu Y.\ \varphi(X, Y)$$

parity automata / vs. alternation depth:



Example

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$
$b \leftarrow \neg c$
$c \leftarrow \neg a$

When is $c(x)$ well-founded?

Example

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$

$b \leftarrow \neg c$

$c \leftarrow \neg a$

When is $c(x)$ well-founded?

"$\mathcal{I}$":

$$x \xrightarrow{r} y \xrightarrow{r} z$$

Start: i) $S = \emptyset$

- $T_{\mathcal{I},\mathcal{C}}(\emptyset) = \emptyset$, $\neg.U_{\mathcal{I},\mathcal{C}}(\emptyset) = \{\neg a(z)\}$

ii) $S = \{\neg a(z)\}$

- $T_{\mathcal{I},\mathcal{C}}(\{\neg a(z)\}) = \{c(z)\}$,
- $\neg.U_{\mathcal{I},\mathcal{C}}(\{\neg a(z)\}) = \{\neg a(z)\}$

iii) $S = \{\neg a(z), c(z)\}$

- $T_{\mathcal{I},\mathcal{C}}(S) = \{c(z)\}$
- $\neg.U_{\mathcal{I},\mathcal{C}}(S) = \{\neg a(z), \neg b(z)\}$

iv) $S = \{\neg a(z), c(z), \neg b(z)\}$

$+ \{\neg a(y)\}$

...

$$\boxed{W_{\mathcal{I},\mathcal{C}}(S) = T_{\mathcal{I},\mathcal{C}}(S) \cup \neg.U_{\mathcal{I},\mathcal{C}}(S)}$$

Example

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$ "$\neg a \leftarrow \forall r.\neg a \vee \forall r.\neg b$"

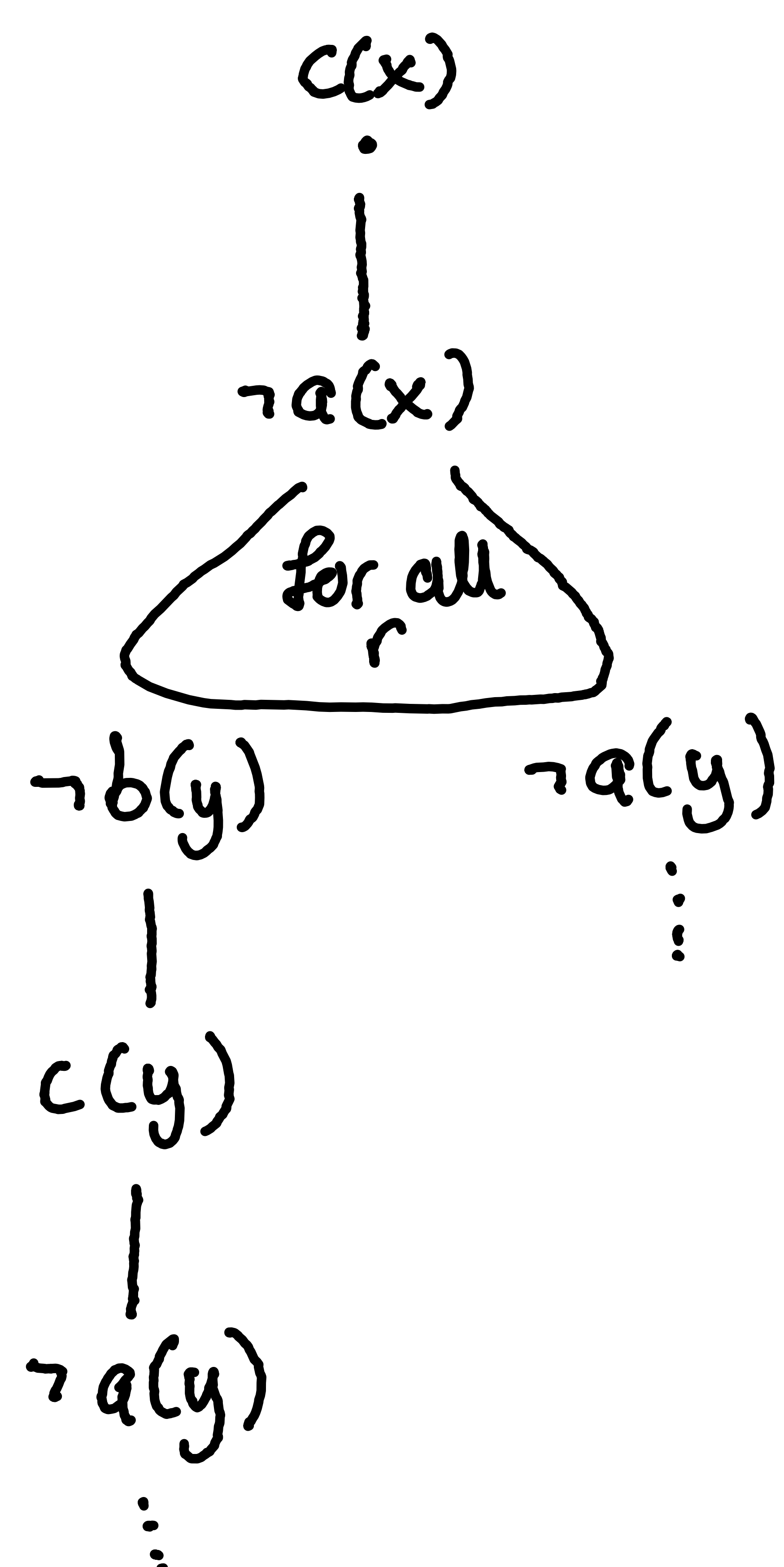
$b \leftarrow \neg c$

$c \leftarrow \neg a$

"$\mathcal{I}$":

$$x \xrightarrow{r} y \xrightarrow{r} z$$

When is $c(x)$ well-founded?



Start: i) $S=\emptyset$

- $T_{\mathcal{I},\mathcal{C}}(\emptyset)=\emptyset$, $\neg.U_{\mathcal{I},\mathcal{C}}(\emptyset)=\{\neg a(z)\}$

ii) $S=\{\neg a(z)\}$

- $T_{\mathcal{I},\mathcal{C}}(\{\neg a(z)\})=\{c(z)\}$,
- $\neg.U_{\mathcal{I},\mathcal{C}}(\{\neg a(z)\})=\{\neg a(z)\}$

iii) $S=\{\neg a(z), c(z)\}$

- $T_{\mathcal{I},\mathcal{C}}(S)=\{c(z)\}$
- $\neg.U_{\mathcal{I},\mathcal{C}}(S)=\{\neg a(z), \neg b(z)\}$

iv) $S=\{\neg a(z), c(z), \neg b(z)\}$
$+ \{\neg a(y)\}$

...

$$\boxed{W_{\mathcal{I},\mathcal{C}}(S) = T_{\mathcal{I},\mathcal{C}}(S) \cup \neg.U_{\mathcal{I},\mathcal{C}}(S)}$$

Example

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$ "$\neg a \leftarrow \forall r.\neg a \vee \forall r.\neg b$"

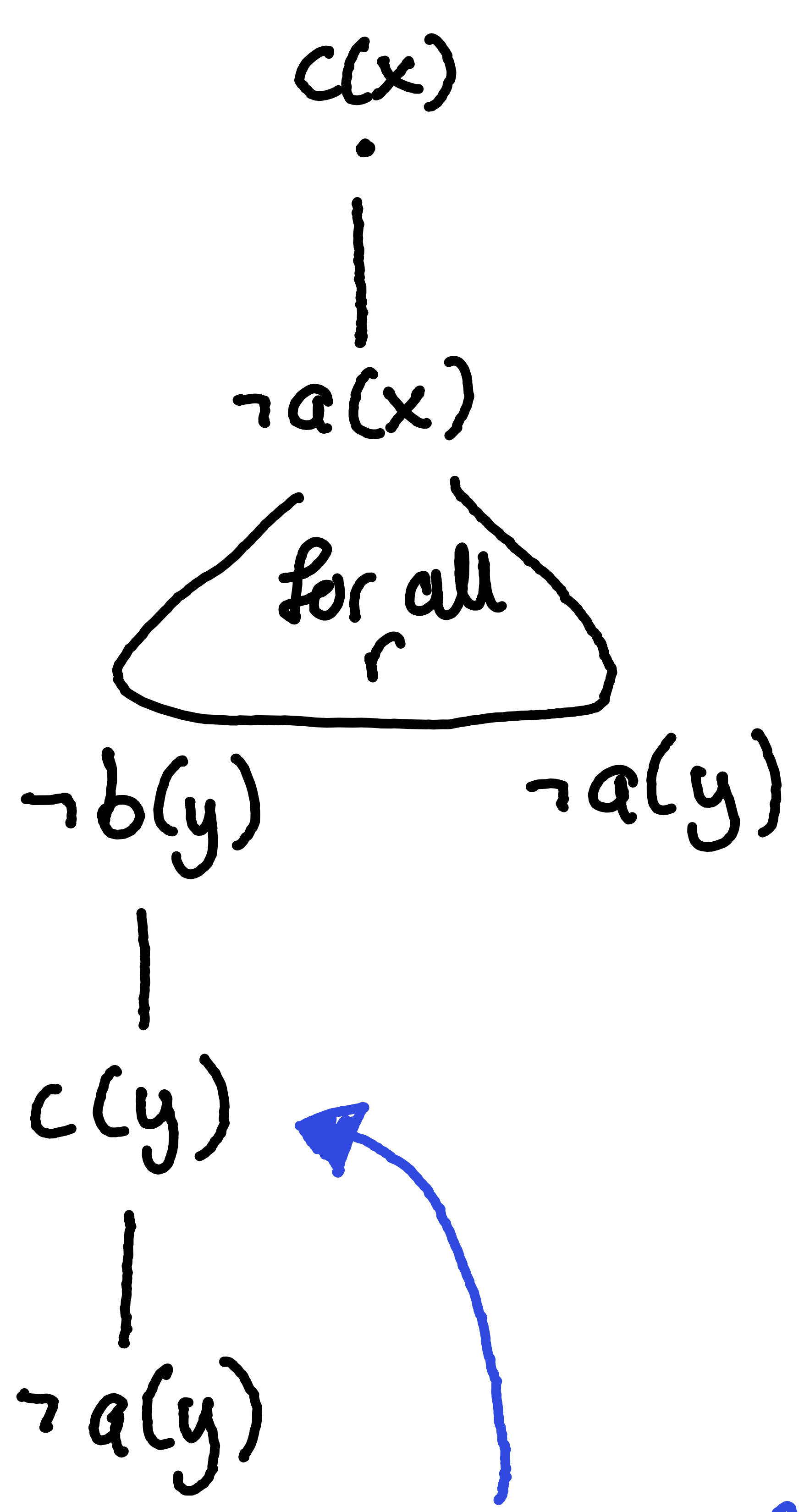
$b \leftarrow \neg c$

$c \leftarrow \neg a$

"$\mathcal{I}$":

$x \xrightarrow{r} y \xrightarrow{r} z$

When is $c(x)$ well-founded?



When all r-paths are finite.

Start: i) $S = \emptyset$

- $T_{\mathcal{I},\mathcal{C}}(\emptyset) = \emptyset$, $\neg.U_{\mathcal{I},\mathcal{C}}(\emptyset) = \{\neg a(z)\}$

ii) $S = \{\neg a(z)\}$

- $T_{\mathcal{I},\mathcal{C}}(\{\neg a(z)\}) = \{c(z)\}$,
- $\neg.U_{\mathcal{I},\mathcal{C}}(\{\neg a(z)\}) = \{\neg a(z)\}$

iii) $S = \{\neg a(z), c(z)\}$

- $T_{\mathcal{I},\mathcal{C}}(S) = \{c(z)\}$
- $\neg.U_{\mathcal{I},\mathcal{C}}(S) = \{\neg a(z), \neg b(z)\}$

iv) $S = \{\neg a(z), c(z), \neg b(z)\}$
$+ \{\neg a(y)\}$
...

$$\boxed{W_{\mathcal{I},\mathcal{C}}(S) = T_{\mathcal{I},\mathcal{C}}(S) \cup \neg.U_{\mathcal{I},\mathcal{C}}(S)}$$

Translation function

Start from $tr^+_{\emptyset,C}(c)$:

$\mu X_c . tr^+_{\{c\},C}(\neg a)$

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$

$b \leftarrow \neg c$

$c \leftarrow \neg a$

$$
\begin{array}{ll}
tr^+_{S,\mathcal{C}}(s \wedge s') := tr^+_{S,\mathcal{C}}(s) \wedge tr^+_{S,\mathcal{C}}(s') & tr^-_{S,\mathcal{C}}(s \wedge s') := tr^-_{S,\mathcal{C}}(s) \vee tr^-_{S,\mathcal{C}}(s') \\
tr^+_{S,\mathcal{C}}(s \vee s') := tr^+_{S,\mathcal{C}}(s) \vee tr^+_{S,\mathcal{C}}(s') & tr^-_{S,\mathcal{C}}(s \vee s') := tr^-_{S,\mathcal{C}}(s) \wedge tr^-_{S,\mathcal{C}}(s') \\
tr^+_{S,\mathcal{C}}(\exists r.s) := \langle r \rangle tr^+_{S,\mathcal{C}}(s) & tr^-_{S,\mathcal{C}}(\exists r.s) := [r] tr^-_{S,\mathcal{C}}(s) \\
tr^+_{S,\mathcal{C}}(\forall r.s) := [r] tr^+_{S,\mathcal{C}}(s) & tr^-_{S,\mathcal{C}}(\forall r.s) := \langle r \rangle tr^-_{S,\mathcal{C}}(s) \\
tr^+_{S,\mathcal{C}}(A) := A \quad tr^+_{S,C}(a) := a & tr^-_{S,\mathcal{C}}(A) := \neg A \quad tr^-_{S,C}(a) := \neg a \\
tr^+_{S,\mathcal{C}}(\neg s) := tr^-_{S,\mathcal{C}}(s) & tr^-_{S,\mathcal{C}}(\neg s) := tr^+_{\underline{pos(S)},\mathcal{C}}(s)
\end{array}
$$

$$
\underline{tr^+_{S,\mathcal{C}}(s)} := \begin{cases} X_s & \text{if } s \in S \\ \underline{\mu X_s . tr^+_{S \cup \{s\},\mathcal{C}}(\varphi)} & \text{if } s \notin S,\ s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

$$
tr^-_{S,\mathcal{C}}(s) := \begin{cases} X_{\bar{s}} & \text{if } \bar{s} \in S \\ \nu X_{\bar{s}} . tr^-_{S \cup \{\bar{s}\},\mathcal{C}}(\varphi) & \text{if } \bar{s} \notin S,\ s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

! Ensures we are in Σ^μ_2

Table 2: Translation functions.

Translation function

Start from $tr^{+}_{\emptyset,\mathcal{C}}(c)$:

$\mu X_c . tr^{-}_{\{c\},\mathcal{C}}(a)$

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$

$b \leftarrow \neg c$

$c \leftarrow \neg a$

$$
\begin{array}{ll}
tr^{+}_{S,\mathcal{C}}(s \wedge s') := tr^{+}_{S,\mathcal{C}}(s) \wedge tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \wedge s') := tr^{-}_{S,\mathcal{C}}(s) \vee tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(s \vee s') := tr^{+}_{S,\mathcal{C}}(s) \vee tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \vee s') := tr^{-}_{S,\mathcal{C}}(s) \wedge tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(\exists r.s) := \langle r \rangle tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\exists r.s) := [r] tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(\forall r.s) := [r] tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\forall r.s) := \langle r \rangle tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(A) := A & tr^{-}_{S,\mathcal{C}}(A) := \neg A \\
\underline{tr^{+}_{S,\mathcal{C}}(\neg s) := tr^{-}_{S,\mathcal{C}}(s)} & tr^{-}_{S,\mathcal{C}}(\neg s) := tr^{+}_{\underline{pos(S)},\mathcal{C}}(s)
\end{array}
$$

! Ensures we are in Σ_2^{μ}

$$
tr^{+}_{S,\mathcal{C}}(s) := \begin{cases} X_s & \text{if } s \in S \\ \mu X_s . tr^{+}_{S \cup \{s\},\mathcal{C}}(\varphi) & \text{if } s \notin S, s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

$$
tr^{-}_{S,\mathcal{C}}(s) := \begin{cases} X_{\bar{s}} & \text{if } \bar{s} \in S \\ \nu X_{\bar{s}} . tr^{-}_{S \cup \{\bar{s}\},\mathcal{C}}(\varphi) & \text{if } \bar{s} \notin S, s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

Table 2: Translation functions.

Translation function

Start from $tr^{+}_{\emptyset,\mathcal{C}}(c)$

$$\mu X_c . \nu X_{\bar{a}} \left([r]\, \underline{tr^{-}_{\{c,\bar{a}\}}(a)} \vee [r]\, \nu X_{\bar{b}} .\, \underline{tr^{-}_{\{c,\bar{a}\}}(\neg c)} \right)$$

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$
$b \leftarrow \neg c$
$c \leftarrow \neg a$

$$
\begin{aligned}
tr^{+}_{S,\mathcal{C}}(s \wedge s') &:= tr^{+}_{S,\mathcal{C}}(s) \wedge tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \wedge s') &:= tr^{-}_{S,\mathcal{C}}(s) \vee tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(s \vee s') &:= tr^{+}_{S,\mathcal{C}}(s) \vee tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \vee s') &:= tr^{-}_{S,\mathcal{C}}(s) \wedge tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(\exists r.s) &:= \langle r \rangle tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\exists r.s) &:= [r] tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(\forall r.s) &:= [r] tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\forall r.s) &:= \langle r \rangle tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(A) &:= A & tr^{-}_{S,\mathcal{C}}(A) &:= \neg A \\
tr^{+}_{S,\mathcal{C}}(\neg s) &:= tr^{-}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\neg s) &:= tr^{+}_{\underline{pos(S)},\mathcal{C}}(s)
\end{aligned}
$$

! Ensures we are in Σ_2^{μ}

$$
tr^{+}_{S,\mathcal{C}}(s) := \begin{cases} X_s & \text{if } s \in S \\ \mu X_s . tr^{+}_{S \cup \{s\},\mathcal{C}}(\varphi) & \text{if } s \notin S, s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

$$
tr^{-}_{S,\mathcal{C}}(s) := \begin{cases} X_{\bar{s}} & \text{if } \bar{s} \in S \\ \nu X_{\bar{s}} . tr^{-}_{S \cup \{\bar{s}\},\mathcal{C}}(\varphi) & \text{if } \bar{s} \notin S, s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

Table 2: Translation functions.

Translation function

Start from $tr^{+}_{\emptyset,\mathcal{C}}(c)$

$$\mu X_c . \nu X_{\bar{a}} \big([r]\, X_{\bar{a}} \quad \vee [r]\, \nu X_{\bar{b}} .\, tr^{+}_{\{c\}}(c) \big)$$

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$

$b \leftarrow \neg c$

$c \leftarrow \neg a$

$$\begin{aligned}
tr^{+}_{S,\mathcal{C}}(s \wedge s') &:= tr^{+}_{S,\mathcal{C}}(s) \wedge tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \wedge s') &:= tr^{-}_{S,\mathcal{C}}(s) \vee tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(s \vee s') &:= tr^{+}_{S,\mathcal{C}}(s) \vee tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \vee s') &:= tr^{-}_{S,\mathcal{C}}(s) \wedge tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(\exists r.s) &:= \langle r \rangle tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\exists r.s) &:= [r] tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(\forall r.s) &:= [r] tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\forall r.s) &:= \langle r \rangle tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(A) &:= A & tr^{-}_{S,\mathcal{C}}(A) &:= \neg A \\
tr^{+}_{S,\mathcal{C}}(\neg s) &:= tr^{-}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\neg s) &:= tr^{+}_{\underline{pos(S)},\mathcal{C}}(s)
\end{aligned}$$

! Ensures we are in Σ^{μ}_{2}

$$tr^{+}_{S,\mathcal{C}}(s) := \begin{cases} X_s & \text{if } s \in S \\ \mu X_s . tr^{+}_{S \cup \{s\},\mathcal{C}}(\varphi) & \text{if } s \notin S,\, s \leftarrow \varphi \in \mathcal{C} \end{cases}$$

$$tr^{-}_{S,\mathcal{C}}(s) := \begin{cases} X_{\bar{s}} & \text{if } \bar{s} \in S \\ \nu X_{\bar{s}} . tr^{-}_{S \cup \{\bar{s}\},\mathcal{C}}(\varphi) & \text{if } \bar{s} \notin S,\, s \leftarrow \varphi \in \mathcal{C} \end{cases}$$

Table 2: Translation functions.

Translation function

Start from $tr^{+}_{\emptyset,\mathcal{C}}(c)$

$\mu X_c . \nu X_{\bar{a}} ([r] X_{\bar{a}} \quad \vee [r] \nu X_{\bar{b}} . X_c)$

$\mathcal{C}$:

$a \leftarrow \exists r.a \wedge \exists r.b$
$b \leftarrow \neg c$
$c \leftarrow \neg a$

$$
\begin{aligned}
tr^{+}_{S,\mathcal{C}}(s \wedge s') &:= tr^{+}_{S,\mathcal{C}}(s) \wedge tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \wedge s') &:= tr^{-}_{S,\mathcal{C}}(s) \vee tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(s \vee s') &:= tr^{+}_{S,\mathcal{C}}(s) \vee tr^{+}_{S,\mathcal{C}}(s') & tr^{-}_{S,\mathcal{C}}(s \vee s') &:= tr^{-}_{S,\mathcal{C}}(s) \wedge tr^{-}_{S,\mathcal{C}}(s') \\
tr^{+}_{S,\mathcal{C}}(\exists r.s) &:= \langle r \rangle tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\exists r.s) &:= [r] tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(\forall r.s) &:= [r] tr^{+}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\forall r.s) &:= \langle r \rangle tr^{-}_{S,\mathcal{C}}(s) \\
tr^{+}_{S,\mathcal{C}}(A) &:= A & tr^{-}_{S,\mathcal{C}}(A) &:= \neg A \\
tr^{+}_{S,\mathcal{C}}(\neg s) &:= tr^{-}_{S,\mathcal{C}}(s) & tr^{-}_{S,\mathcal{C}}(\neg s) &:= tr^{+}_{\underline{pos(S)},\mathcal{C}}(s)
\end{aligned}
$$

! Ensures we are in Σ_2^{μ}

$$
tr^{+}_{S,\mathcal{C}}(s) := \begin{cases} X_s & \text{if } s \in S \\ \mu X_s . tr^{+}_{S \cup \{s\},\mathcal{C}}(\varphi) & \text{if } s \notin S,\, s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

$$
tr^{-}_{S,\mathcal{C}}(s) := \begin{cases} X_{\bar{s}} & \text{if } \bar{s} \in S \\ \nu X_{\bar{s}} . tr^{-}_{S \cup \{\bar{s}\},\mathcal{C}}(\varphi) & \text{if } \bar{s} \notin S,\, s \leftarrow \varphi \in \mathcal{C} \end{cases}
$$

Table 2: Translation functions.

Removing abundant parts

$\mu X_c . \nu X_{\bar{a}} ([r] X_{\bar{a}} \vee [r] \nu X_{\bar{b}} . X_c)$ – to improve readability

$$
\begin{array}{ll}
\mathit{fin}(\varphi \wedge \varphi') := \mathit{fin}(\varphi) \wedge \mathit{fin}(\varphi') & \mathit{fin}(\varphi \vee \varphi') := \mathit{fin}(\varphi) \vee \mathit{fin}(\varphi') \\
\mathit{fin}(\langle r \rangle \varphi) := \langle r \rangle \mathit{fin}(\varphi) & \mathit{fin}([r]\varphi) := [r]\mathit{fin}(\varphi) \\
\mathit{fin}(\neg \varphi) := \neg \mathit{fin}(\varphi) & \mathit{fin}(A) := A \\
\mathit{fin}(X) := X & \boxed{\mathit{fin}(\sigma X.\varphi) := \begin{cases} \mathit{fin}(\varphi) & \text{if } X \in sub(\varphi) \\ \sigma X.\mathit{fin}(\varphi) & \text{if } X \notin sub(\varphi) \end{cases}}
\end{array}
$$

Table 3: Finalising function.

Removing abundant parts

$$\mu X_c . \nu X_{\bar a}([r]\, X_{\bar a} \vee [r]\, \nu X_{\bar b} . X_c)$$

$$\Big\downarrow \ \text{fin-function}$$

$$\mu X_c . \nu X_{\bar a}.([r] X_{\bar a} \vee [r] X_c) \qquad (\nu X. [r] X \equiv \top)$$

$$\equiv$$

$$\mu X_c . [r] X_c$$ – "all r-paths are finite"

$$
\begin{array}{ll}
\mathit{fin}(\varphi \wedge \varphi') := \mathit{fin}(\varphi) \wedge \mathit{fin}(\varphi') & \mathit{fin}(\varphi \vee \varphi') := \mathit{fin}(\varphi) \vee \mathit{fin}(\varphi') \\
\mathit{fin}(\langle r \rangle \varphi) := \langle r \rangle \mathit{fin}(\varphi) & \mathit{fin}([r]\varphi) := [r]\mathit{fin}(\varphi) \\
\mathit{fin}(\neg \varphi) := \neg \mathit{fin}(\varphi) & \mathit{fin}(A) := A \\
\mathit{fin}(X) := X & \boxed{\mathit{fin}(\sigma X.\varphi) := \begin{cases} \mathit{fin}(\varphi) & \text{if } X \in \mathit{sub}(\varphi) \\ \sigma X.\mathit{fin}(\varphi) & \text{if } X \notin \mathit{sub}(\varphi) \end{cases}}
\end{array}
$$

Table 3: Finalising function.

Alternation degree

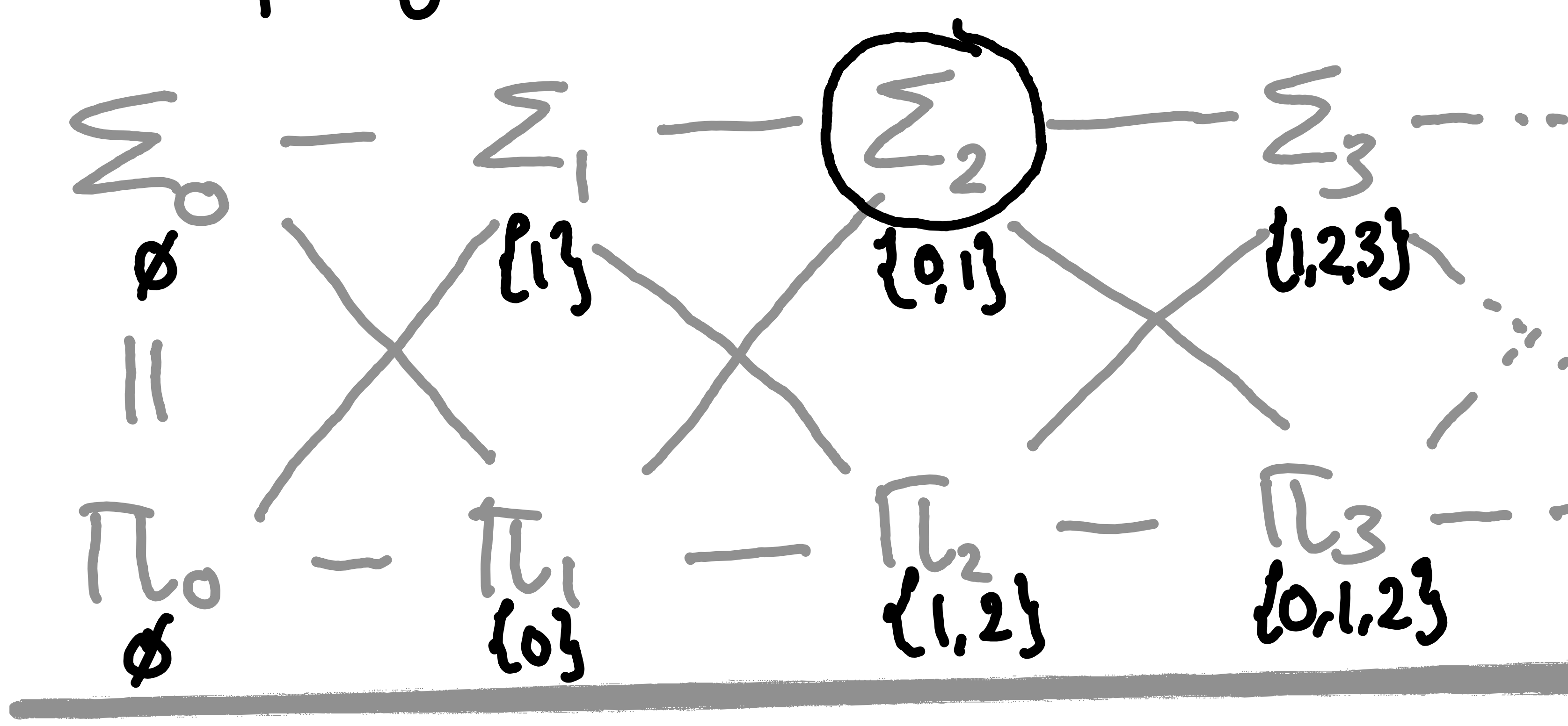
$\mu X. [r] X$ ← no/one alternation

has more alternation →

• Note the least- and greatest- fixed points already showing up...

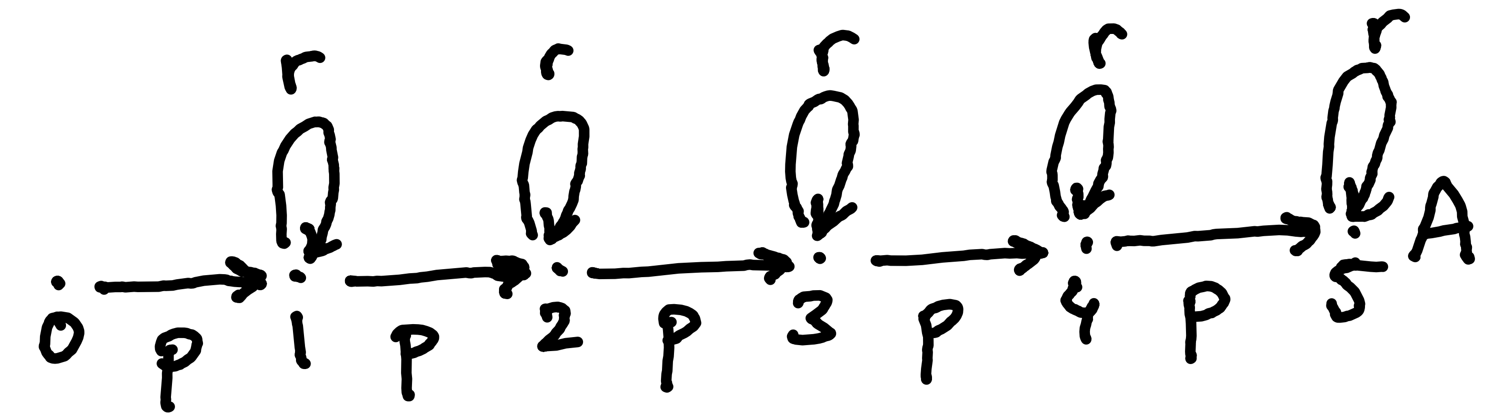
$$\mu X \ldots \nu Y.\ \varphi(X, Y)$$

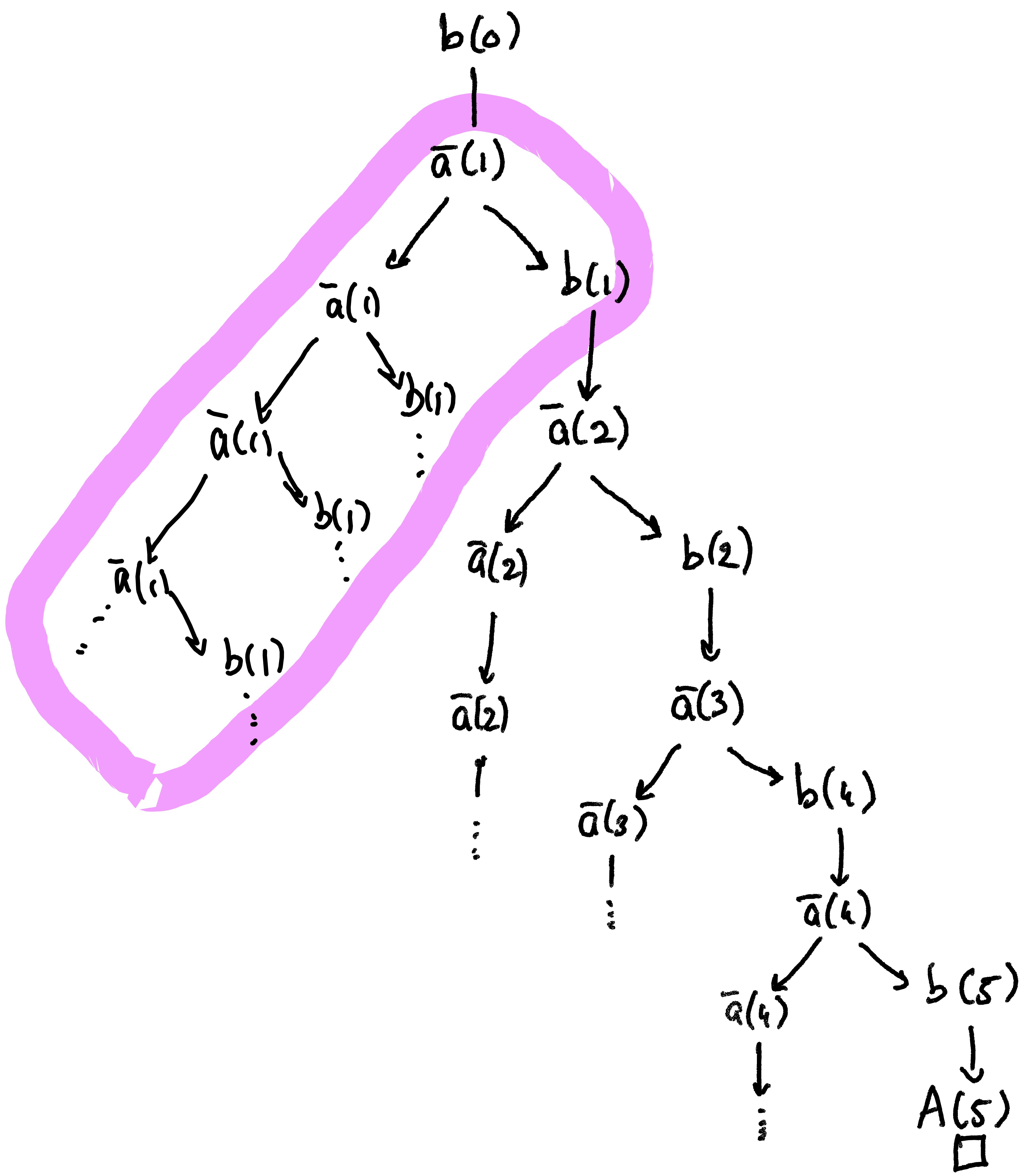
parity automata / vs. alternation depth:



Alternation degree

"I":

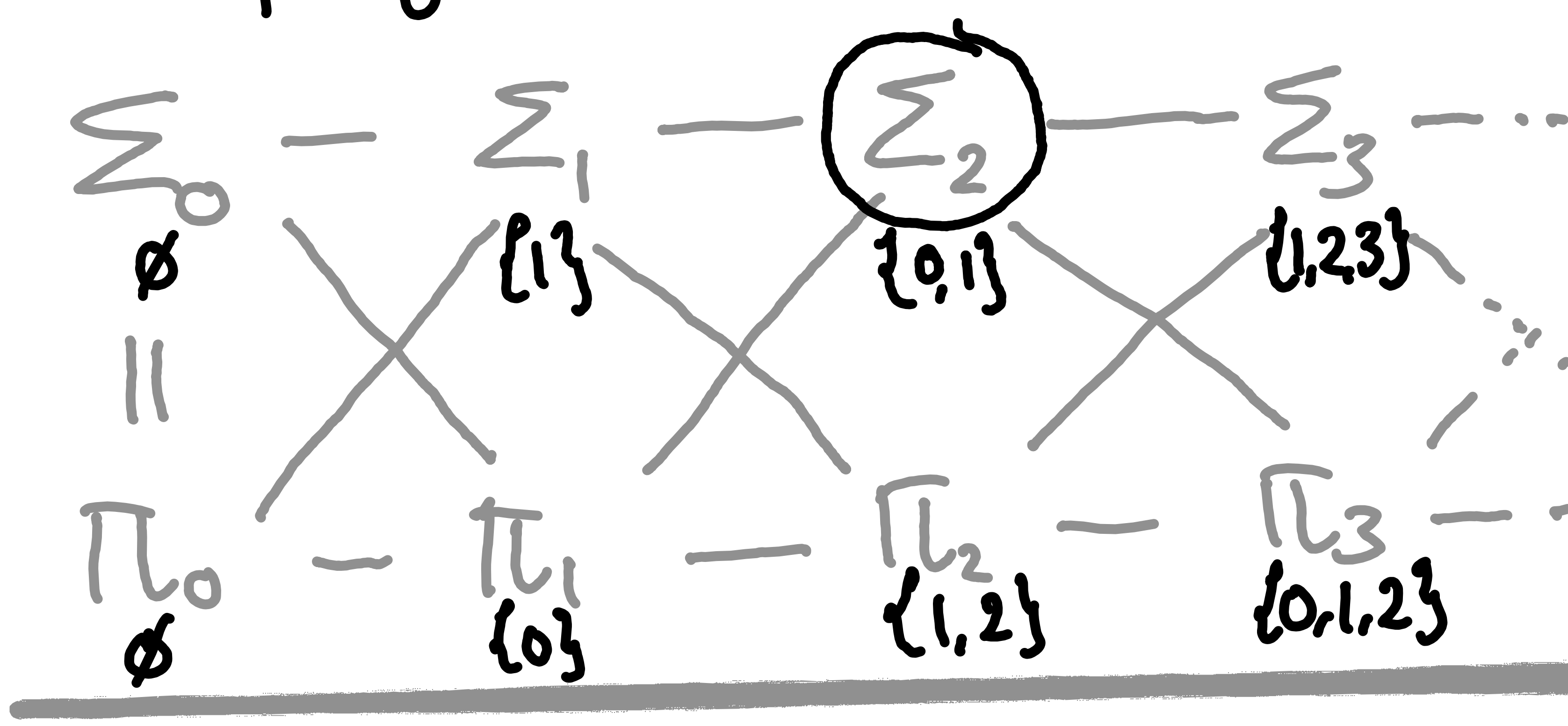




- Note the least- and greatest- fixed points already showing up...

$$\mu X \ldots \nu Y.\ \varphi(X, Y)$$

parity automata/vs. alternation depth:



$a \leftarrow \neg b \vee \exists r.a$ "$\neg a \leftarrow b \wedge \forall r.\neg a$"

$b \leftarrow A \vee \exists p.\neg a$

When is b(x) well-founded?

$$\mu X_b.(A \vee \langle p \rangle \nu X_{\bar{a}}(X_b \wedge [r] X_{\bar{a}}))$$

Final translation

- For each shapes graph $(\mathcal{C}, \{s(a)\})$, $\mathcal{I}$ validates $(\mathcal{C}, \{s(a)\})$ iff there exists $v \in \Delta^{\mathcal{I}}$ s.t. $\mathcal{I}, v \models \mathit{fin}(tr^{+}_{\emptyset, \mathcal{C}}(s))$

Final translation

- For each shapes graph $(\mathcal{C}, \{s(a)\})$, $\mathcal{I}$ validates $(\mathcal{C}, \{s(a)\})$ iff there exists $v \in \Delta^{\mathcal{I}}$ s.t. $\mathcal{I}, v \models \mathit{fin}(tr^{+}_{\emptyset,\mathcal{C}}(s))$ — extendable to full documents.

- Document implication: $\quad \Theta_{\mathcal{C},T}$: conjunction of $\begin{cases} \neg a \vee tr^{+}_{\emptyset,\mathcal{C}}(s) & (a,s) \in T \\ \neg A \vee tr^{+}_{\emptyset,\mathcal{C}}(s) & (A,s) \in T \\ \neg\langle r\rangle \vee tr^{+}_{\emptyset,\mathcal{C}}(s) & (r,s) \in T \end{cases}$

Then $(\mathcal{C},T)$ (finitely) implies $(\mathcal{C}',T')$ iff

$$\bigwedge_{a \in \mathcal{I}} \langle p\rangle(a \wedge \Lambda_{\mathcal{C},T,p}) \wedge \langle p\rangle \neg \Lambda_{\mathcal{C}',T',p}$$

individuals in $\mathcal{C},T,\mathcal{C}'$ & T'

fresh role p

where

$$\Lambda_{\mathcal{C},T,p} := \nu X.\left(\Theta_{\mathcal{C},T} \wedge \bigwedge_{r \in R \cup \{p\}} \left([r^-]X \wedge [r]X\right)\right)$$

roles appearing in $(\mathcal{C},T)$

Complexity Results

- Then: satisfiability of $\mathcal{ALCOI}_s^{wf} \overset{*}{\rightsquigarrow}$ modal μ-calculus satisfiability (EXPTIME-c)

  containment of $\mathcal{ALCOI}_s^{wf} \overset{*}{\rightsquigarrow}$ is $fn(tr^+_{\phi,e}(s)) \wedge \neg fn(tr^+_{\phi,e'}(s'))$ sat.? (EXPTIME-c)

  finite satisfiability of $\mathcal{ALCOI}_s^{wf} \rightsquigarrow$ decidable (in 3-EXPTIME)

  for $\mathcal{ALCO}_s^{wf}$: EXPTIME-C

*: translation may cause exponential blow-up but with repetition of similar formulas – direct translation into two-way alternating parity tree automata for tight bound

Summary

- Supported model semantics:
  - Satisfiability similar to DL satisfiability.
  - Containment undecidable for $\mathcal{ALCOI}$s already (not like the DL setting)

- Well-founded semantics:
  - Translation into modal μ-calculus, EXPTIME-c results for both satisfiability & containment (for the $\mathcal{ALCOI}$ fragment)
  - Work in process: counting? More expressive / different fragments?

Thank you for listening!