Abstract Interpretation as Automated Deduction

Vijay D'Silva

Caterina Urban

August 6th, 2015
CADE 2015
Berlin, Germany

bottom line: an **abstract interpreter** can be understood as a **sound** but *incomplete* **solver** for monadic second order logic extended with a first-order theory

$$0.0 \leq x \land x \leq 10.0 \land y = x^5 \land y > 10^5$$

$$x \mapsto (-\infty, +\infty) \qquad y \mapsto (-\infty, +\infty)$$

$$x \mapsto [0.0, 10.0] \qquad y \mapsto (10^5, +\infty)$$

$$x \mapsto [0.0, 10.0] \qquad y \mapsto (10^5, 10^5]$$

$$z = y \land x = y \cdot z \land x < 0$$

$$x \mapsto (-\infty, +\infty) \qquad y \mapsto (-\infty, +\infty) \qquad z \mapsto (-\infty, +\infty)$$

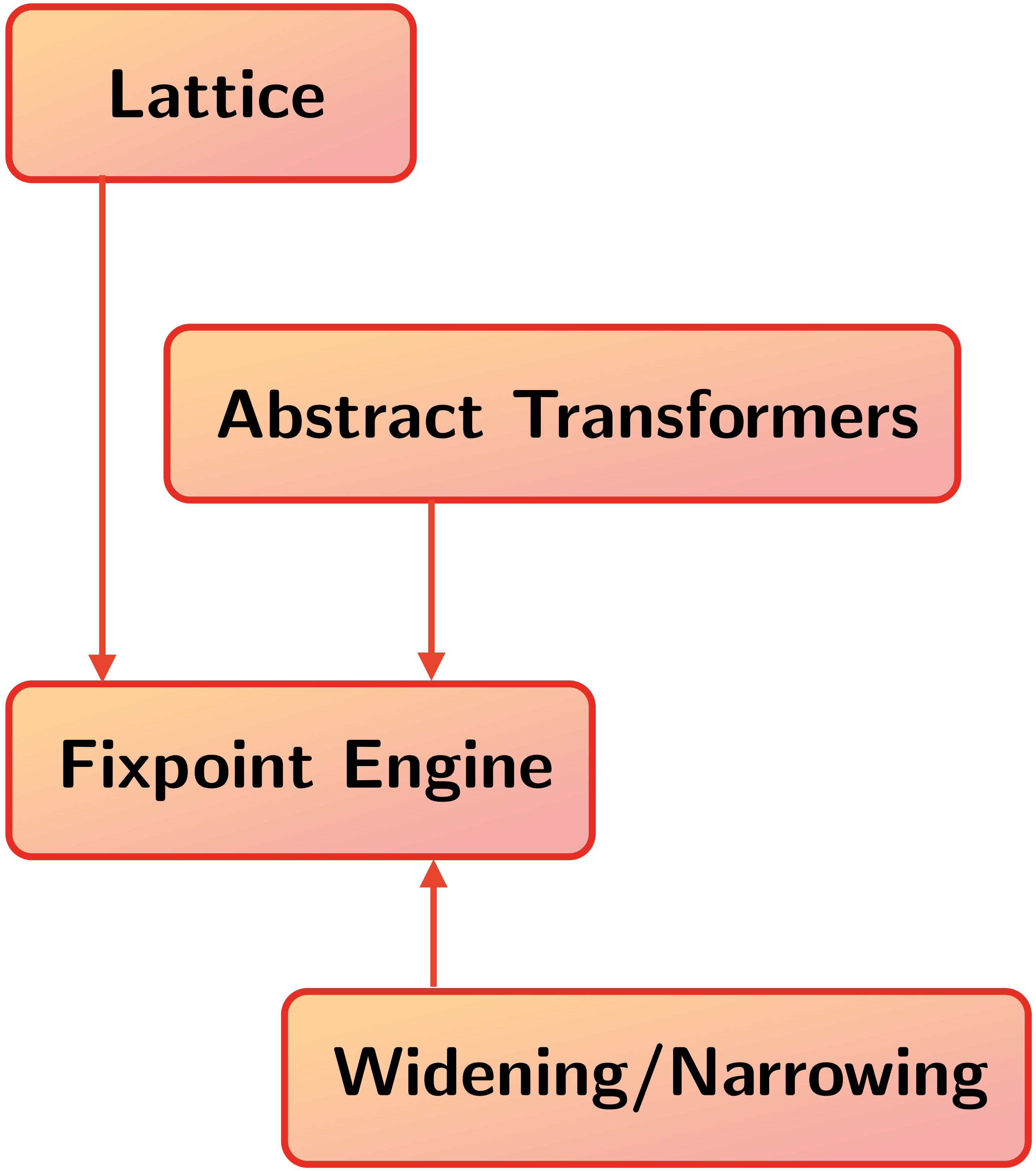
$$x \mapsto (-\infty, 0) \qquad y \mapsto (-\infty, +\infty) \qquad z \mapsto (-\infty, +\infty)$$

the analysis is **sound**. . .

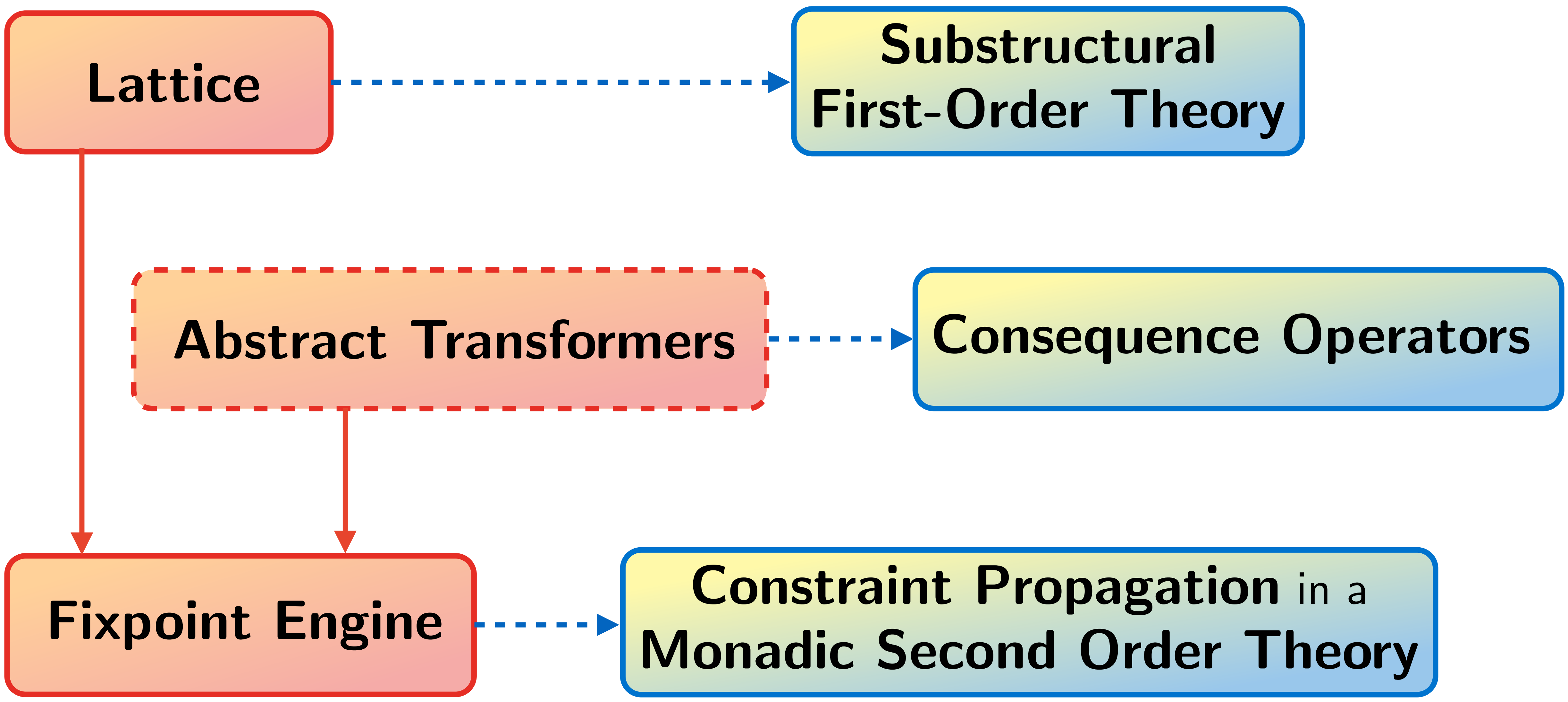
. . . but **incomplete**

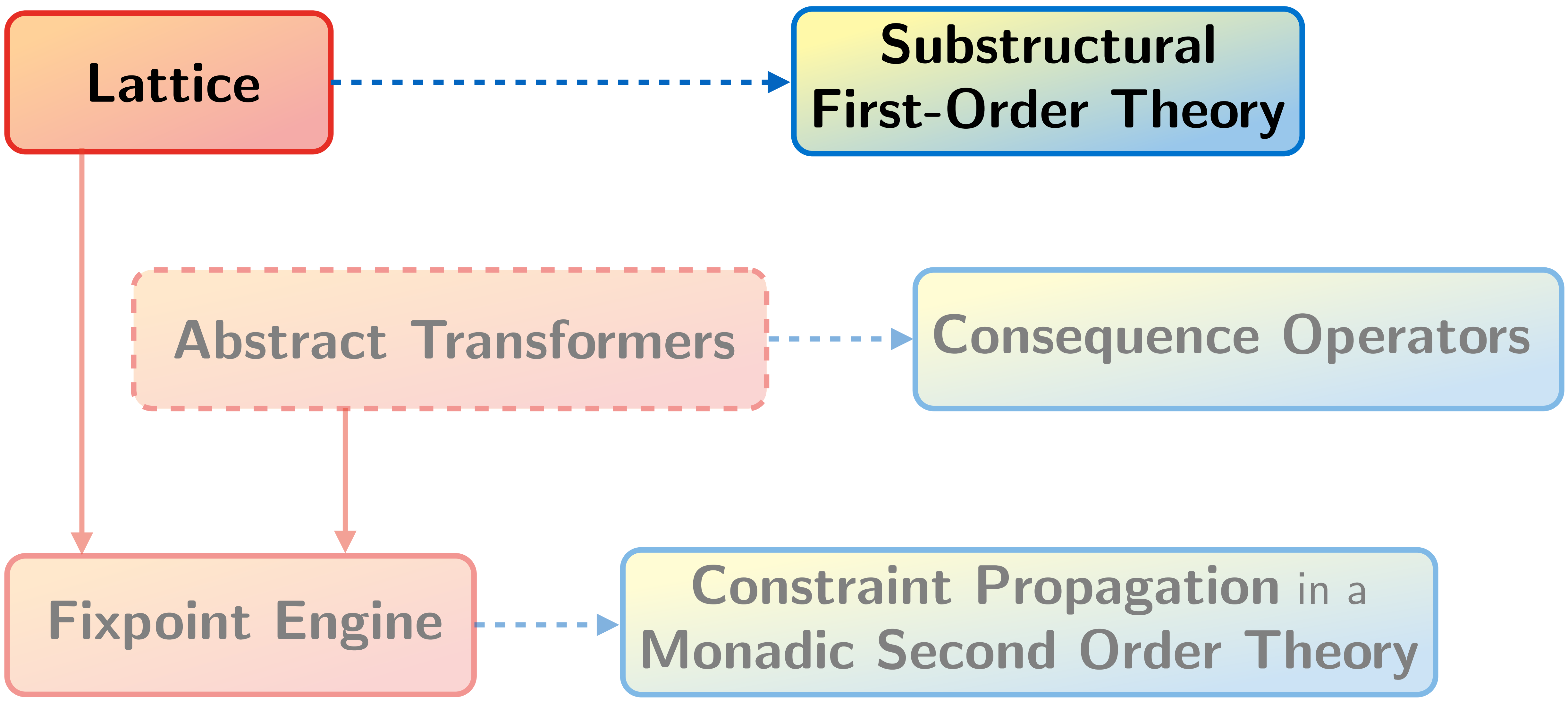
Brain & D. & Griggio & Haller & Kroening - Deciding Floating-Point Logic with Abstract Conflict Driven Clause Learning (FMCAD 2014)

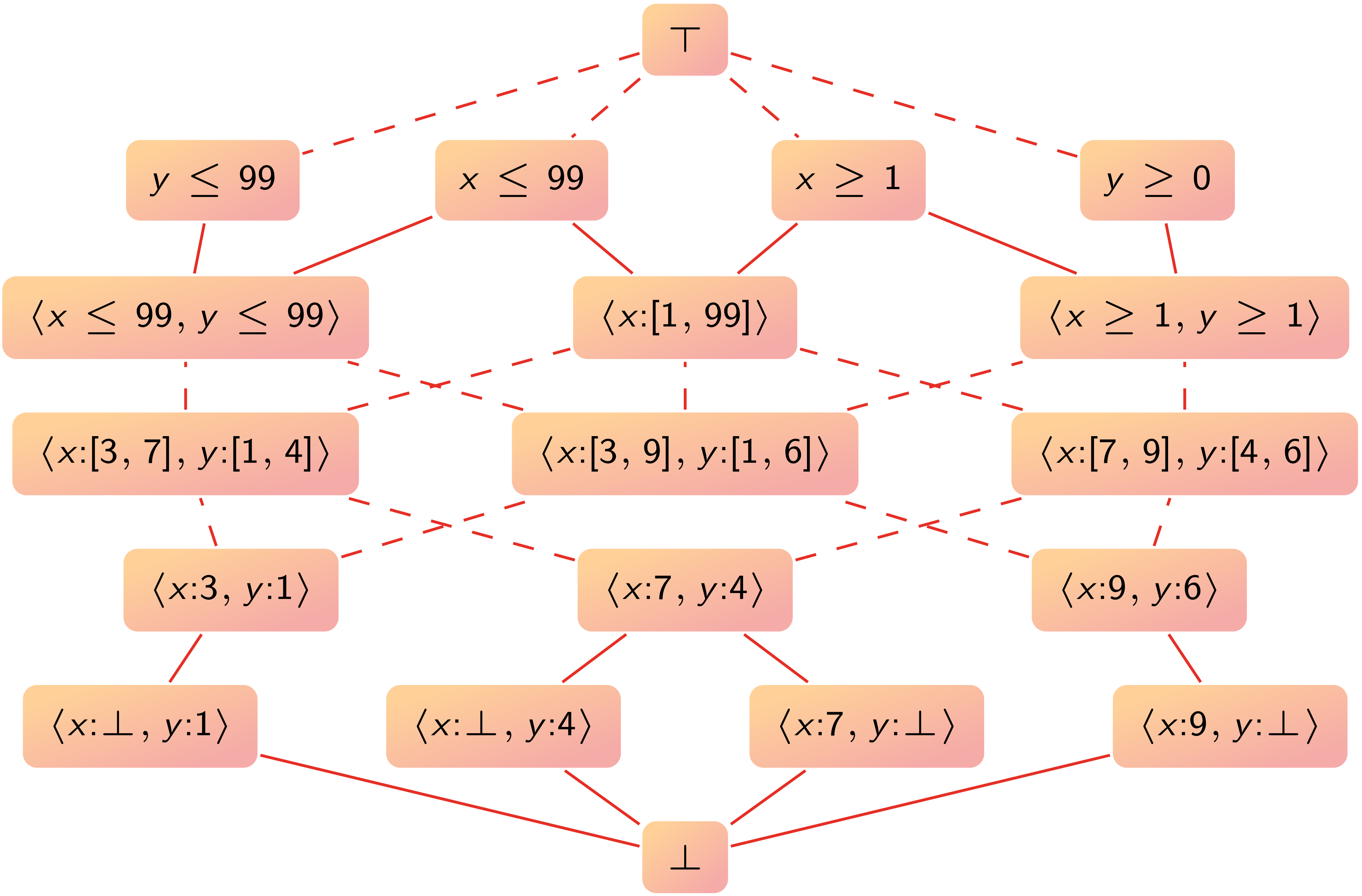
- **value approximation**
- **approximate reasoning**
- **performance** improvement
- systematic way to develop specialized solvers when general solvers are not available



- how can we generate proofs when an abstract interpreter is used in a decision procedure?
- can abstract interpreters be modified to generate a proof certificate that can be checked independently?
- is there a mathematical framework to aid in incorporating ideas from SMT solvers in abstract interpreters?

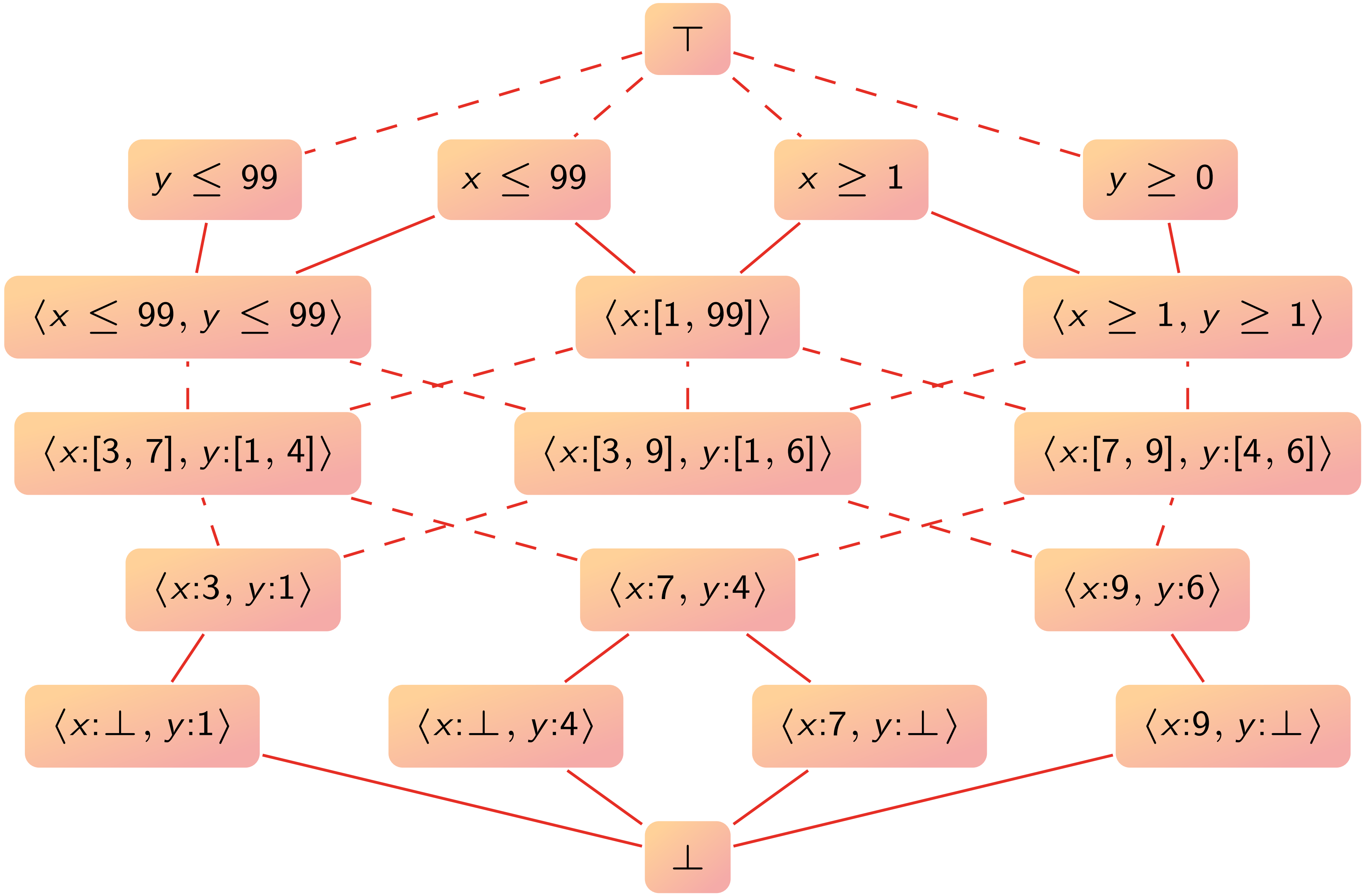

Lattice
Substructural
First-Order Theory
Abstract Transformers
Consequence Operators
Fixpoint Engine
Constraint Propagation in a
Monadic Second Order Theory


Lattice
Substructural First-Order Theory
Abstract Transformers
Consequence Operators
Fixpoint Engine
Constraint Propagation in a Monadic Second Order Theory

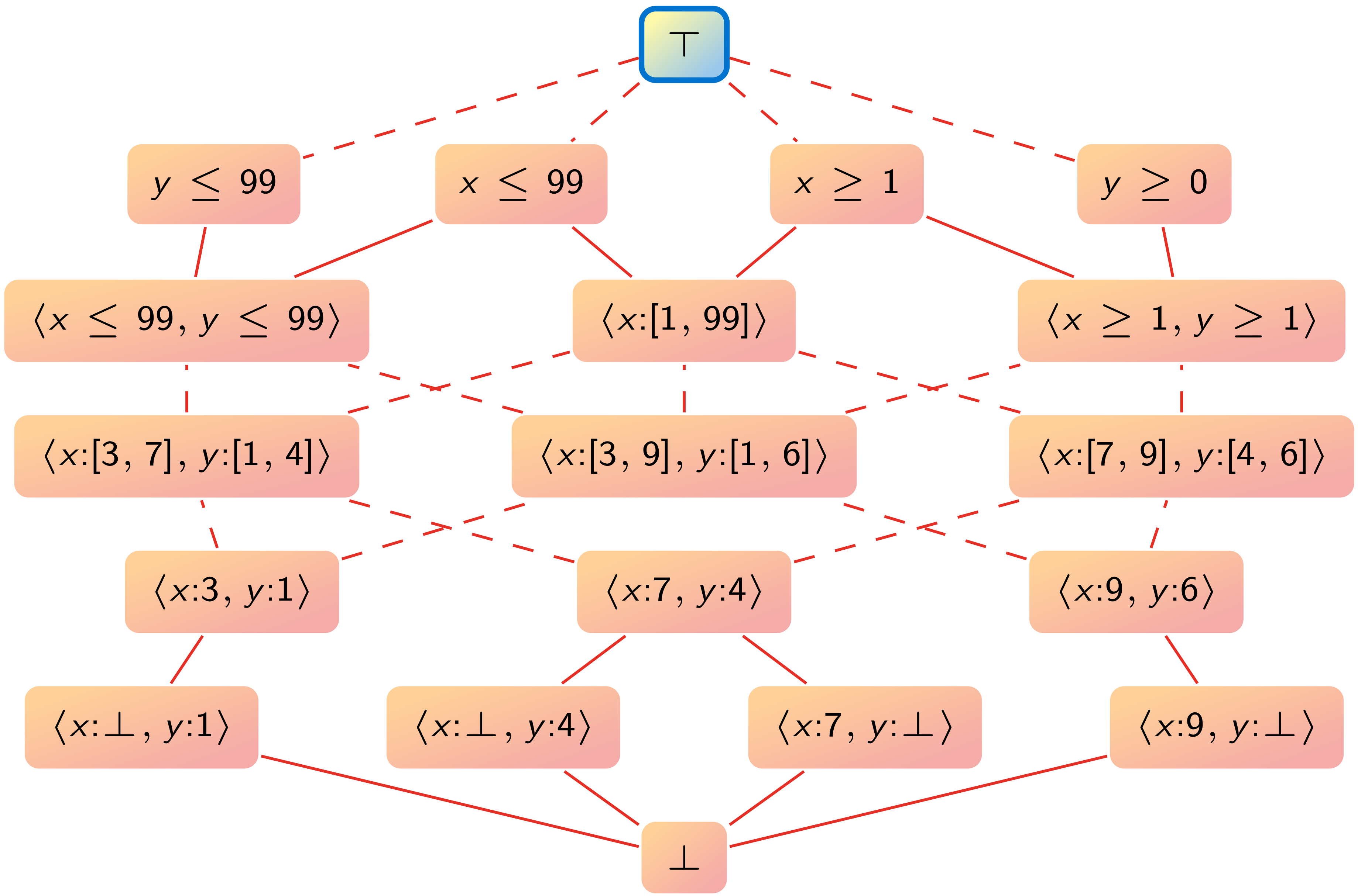


- what are the **formulae** of the logic?
- what is the **proof system** of the logic?
- how can we prove that the logic captures the lattice?

- what are the **formulae** of the logic?

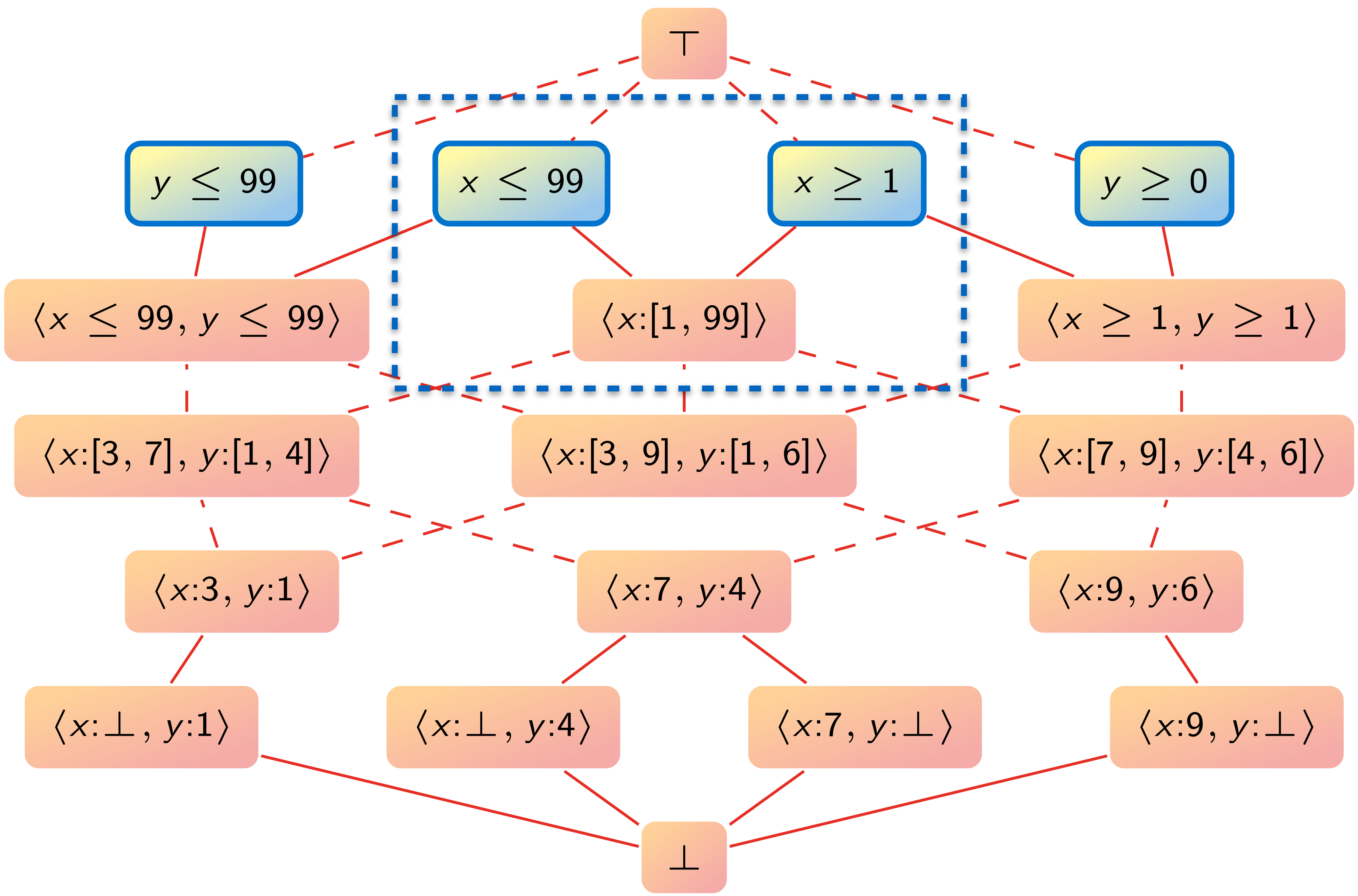


- what are the **formulae** of the logic?



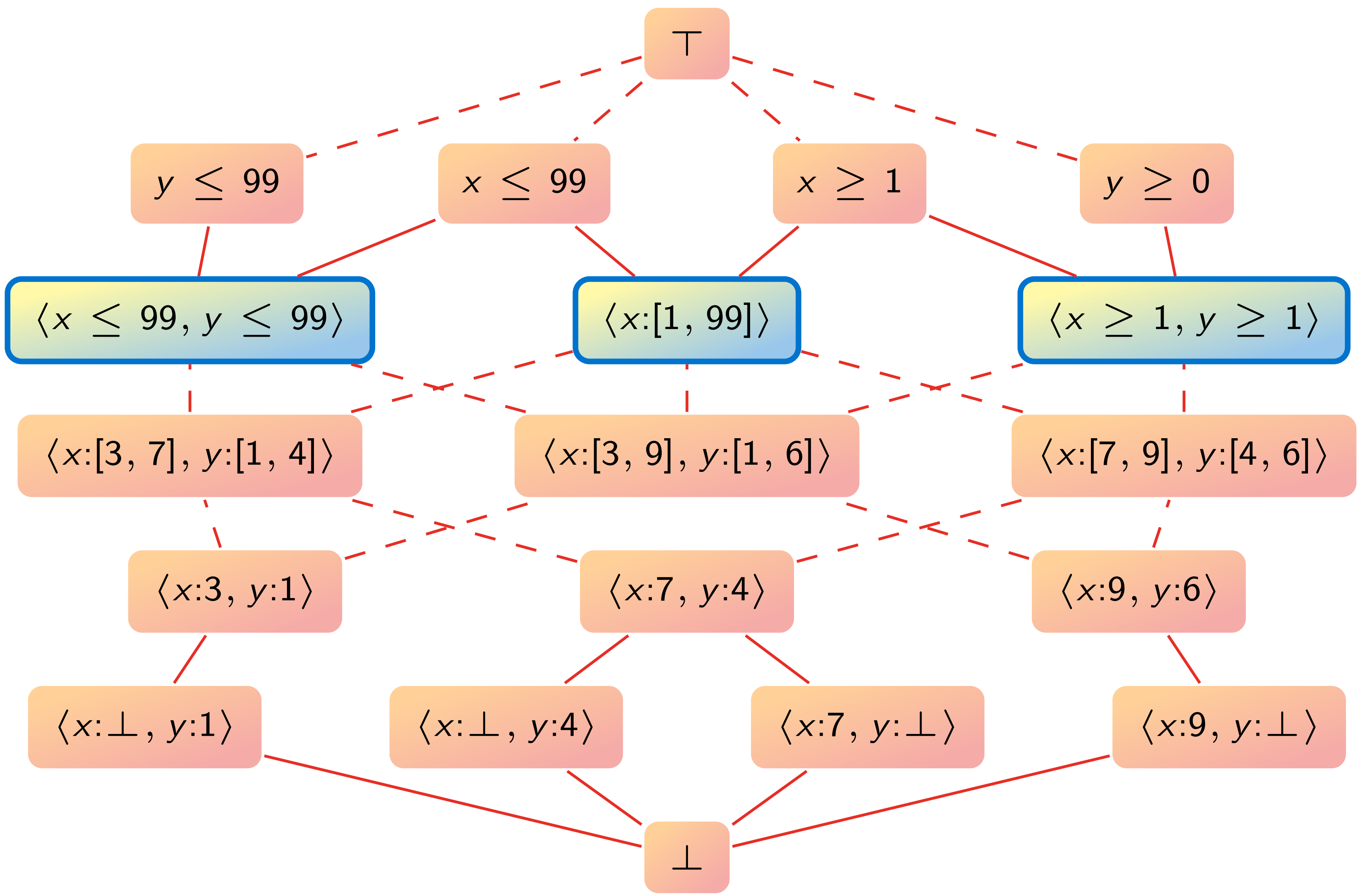
$$\varphi \quad ::= \quad \mathsf{tt}$$

- what are the **formulae** of the logic?



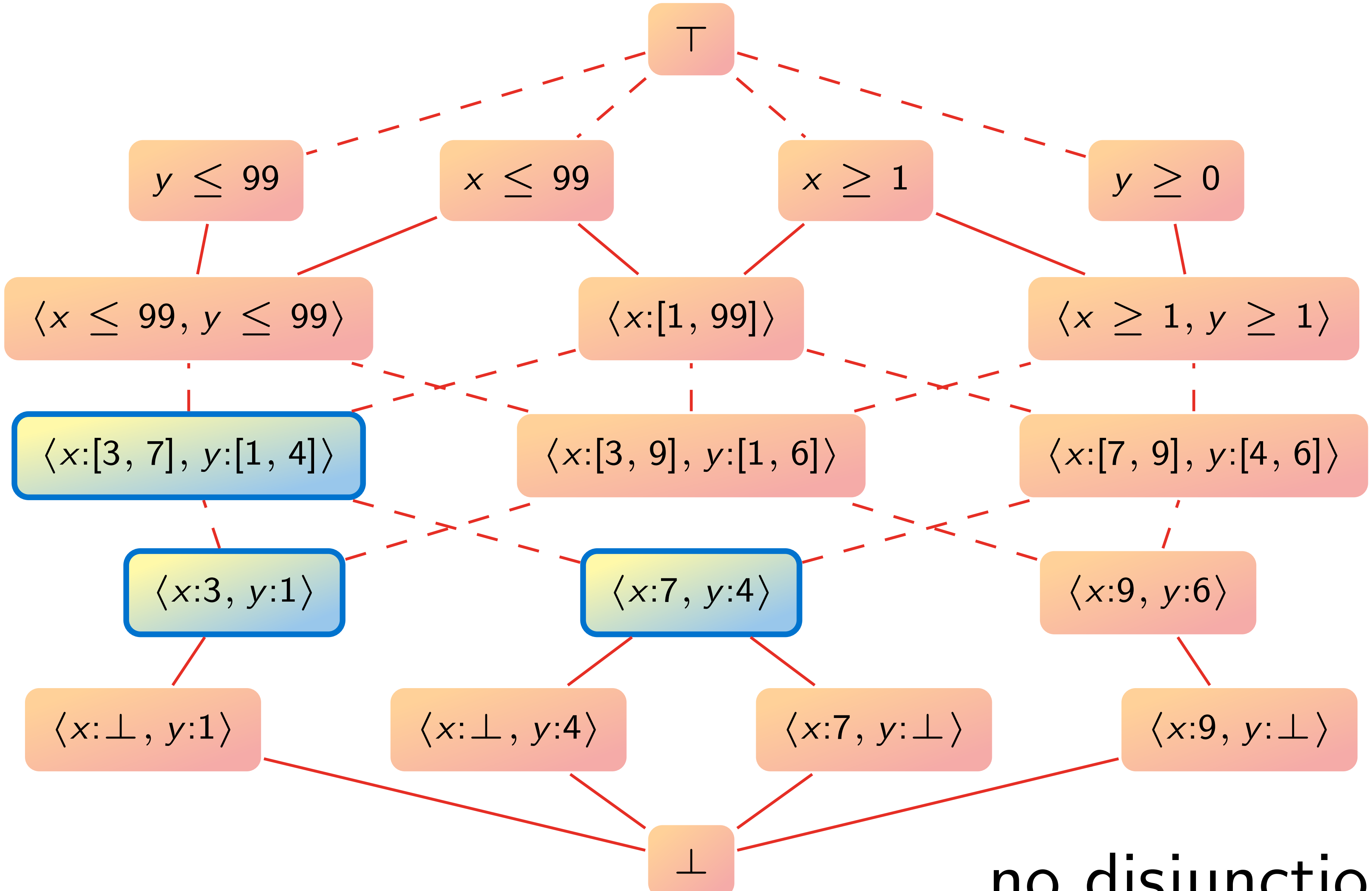
$$\varphi \ ::= \ \mathrm{tt} \mid x \leq k \mid x \geq k$$

- what are the **formulae** of the logic?



$$\varphi \ ::= \ \mathrm{tt} \mid x \leq k \mid x \geq k \mid \varphi \wedge \varphi$$

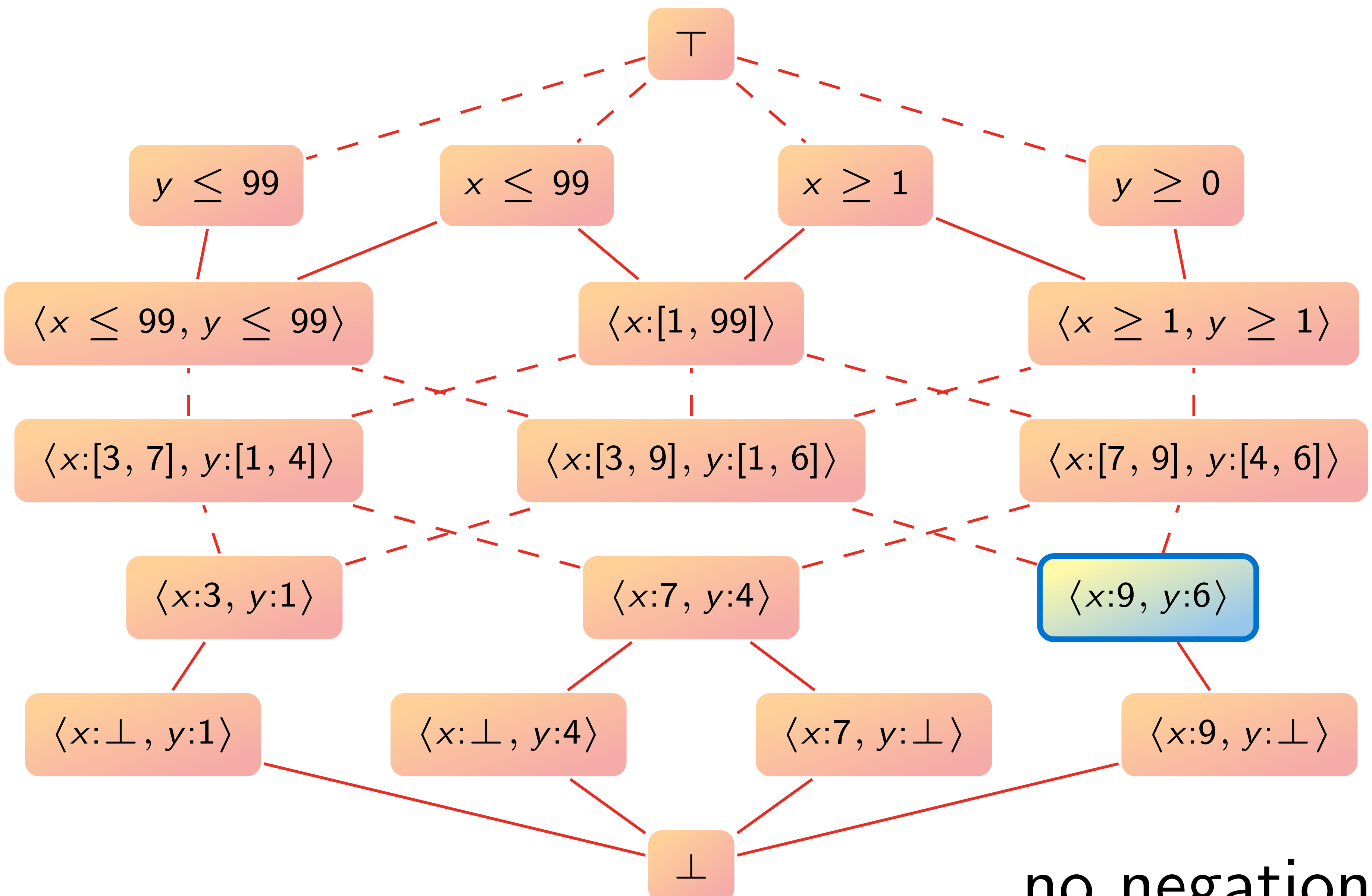
- what are the **formulae** of the logic?



no disjunction

$$\varphi \ ::= \ \mathrm{tt} \mid x \leq k \mid x \geq k \mid \varphi \wedge \varphi$$

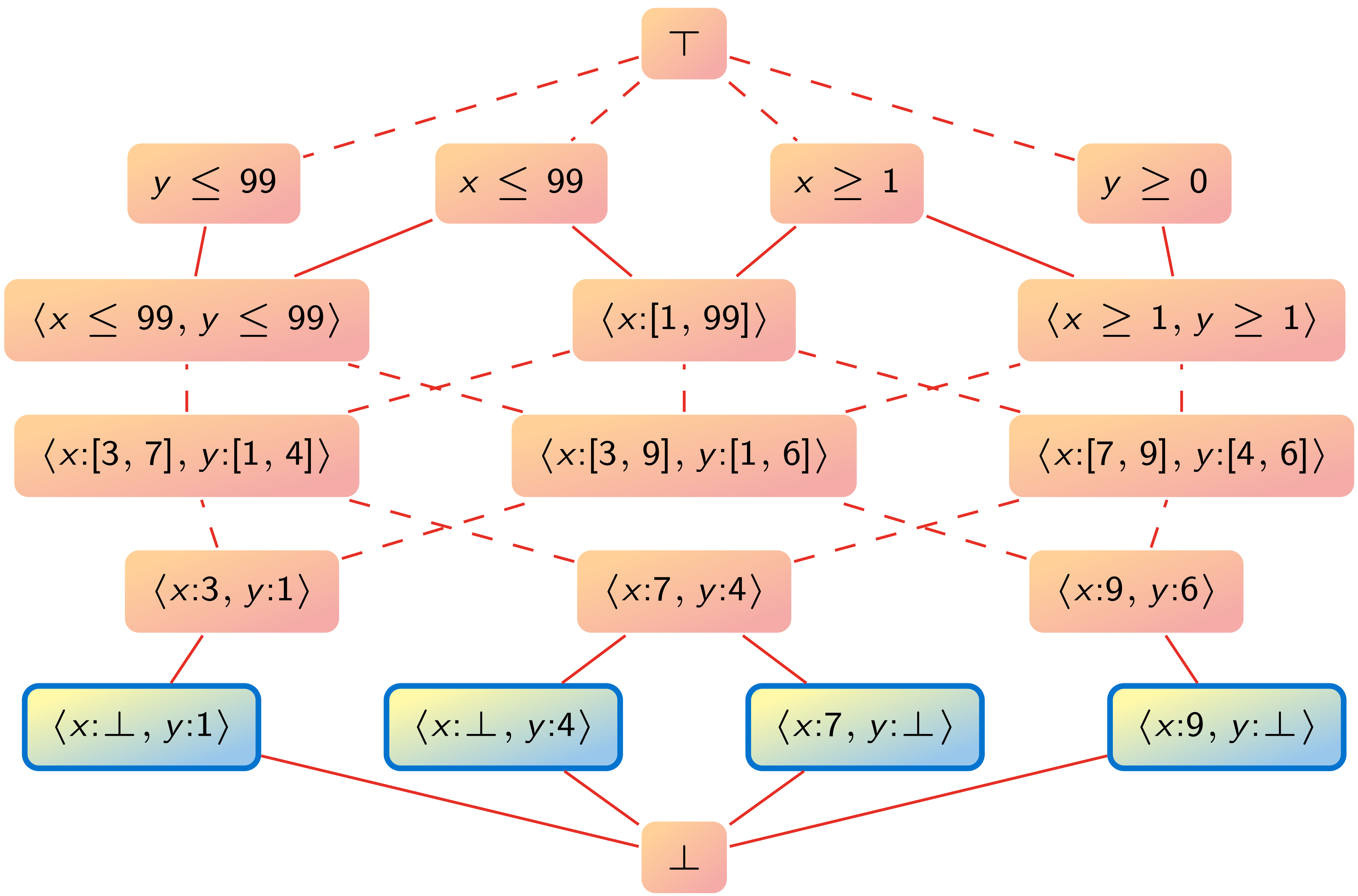
- what are the **formulae** of the logic?



no negation

$$\varphi \ ::= \ \mathrm{tt} \mid x \leq k \mid x \geq k \mid \varphi \wedge \varphi$$

- what are the **formulae** of the logic?



$$\varphi \ ::= \ \mathsf{tt} \mid x \leq k \mid x \geq k \mid \varphi \wedge \varphi \mid \mathsf{ff}_x$$

- what is the **proof system** of the logic?

$\Gamma, \Sigma \vdash \Delta, \Theta$ standard Gentzen sequent

standard interpretation $\Gamma \wedge \Sigma \Rightarrow \Delta \vee \Theta$

$$\Gamma, \Sigma \vdash \varphi$$

single first-order formula

substructural logic

- what is the **proof system** of the logic?

$$\frac{}{\Gamma \vdash \mathsf{tt}}\ \mathsf{ttR} \qquad \frac{}{\mathsf{ff}_x \vdash \varphi(x)}\ \mathsf{ffL}$$

$$\begin{array}{lcl} \mathsf{ff}_x & \vdash & x \geq 5 \wedge x \leq 10 \\ \mathsf{ff}_x & \nvdash & x \geq 5 \wedge y \leq 1 \end{array}$$

- what is the **proof system** of the logic?

$$\frac{}{\Gamma \vdash \mathsf{tt}}\ \text{ttR} \qquad \frac{}{\mathsf{ff}_x \vdash \varphi(x)}\ \text{ffL}$$

$$\frac{}{\varphi \vdash \varphi}\ \text{I} \qquad \frac{\Gamma \vdash \varphi \qquad \varphi, \Sigma \vdash \psi}{\Gamma, \Sigma \vdash \psi}\ \text{CUT} \qquad \frac{\Gamma \vdash \psi}{\Gamma, \varphi \vdash \psi}\ \text{WL} \qquad \frac{\Gamma, \varphi, \varphi \vdash \psi}{\Gamma, \varphi \vdash \psi}\ \text{CL} \qquad \frac{\Gamma, \varphi, \psi \vdash \theta}{\Gamma, \psi, \varphi \vdash \theta}\ \text{PL}$$

standard structural and cut rules

- what is the **proof system** of the logic?

$$\frac{}{\Gamma \vdash \mathsf{tt}}\ \text{ttR} \qquad \frac{}{\mathsf{ff}_x \vdash \varphi(x)}\ \text{ffL}$$

$$\frac{}{\varphi \vdash \varphi}\ \text{I} \qquad \frac{\Gamma \vdash \varphi \qquad \varphi, \Sigma \vdash \psi}{\Gamma, \Sigma \vdash \psi}\ \text{CUT} \qquad \frac{\Gamma \vdash \psi}{\Gamma, \varphi \vdash \psi}\ \text{WL} \qquad \frac{\Gamma, \varphi, \varphi \vdash \psi}{\Gamma, \varphi \vdash \psi}\ \text{CL} \qquad \frac{\Gamma, \varphi, \psi \vdash \theta}{\Gamma, \psi, \varphi \vdash \theta}\ \text{PL}$$

$$\frac{\Gamma, \varphi \vdash \theta}{\Gamma, \varphi \wedge \psi \vdash \theta}\ \wedge\text{L}_1 \qquad \frac{\Gamma, \psi \vdash \theta}{\Gamma, \varphi \wedge \psi \vdash \theta}\ \wedge\text{L}_2 \qquad \frac{\Gamma \vdash \varphi \qquad \Sigma \vdash \psi}{\Gamma, \Sigma \vdash \varphi \wedge \psi}\ \wedge\text{R}$$

standard logical rules for conjunction

- what is the **proof system** of the logic?

$$\frac{}{\Gamma \vdash \mathsf{tt}}\ \text{ttR} \qquad \frac{}{\mathsf{ff}_x \vdash \varphi(x)}\ \text{ffL}$$

$$\frac{}{\varphi \vdash \varphi}\ \text{I} \qquad \frac{\Gamma \vdash \varphi \qquad \varphi, \Sigma \vdash \psi}{\Gamma, \Sigma \vdash \psi}\ \text{CUT} \qquad \frac{\Gamma \vdash \psi}{\Gamma, \varphi \vdash \psi}\ \text{WL} \qquad \frac{\Gamma, \varphi, \varphi \vdash \psi}{\Gamma, \varphi \vdash \psi}\ \text{CL} \qquad \frac{\Gamma, \varphi, \psi \vdash \theta}{\Gamma, \psi, \varphi \vdash \theta}\ \text{PL}$$

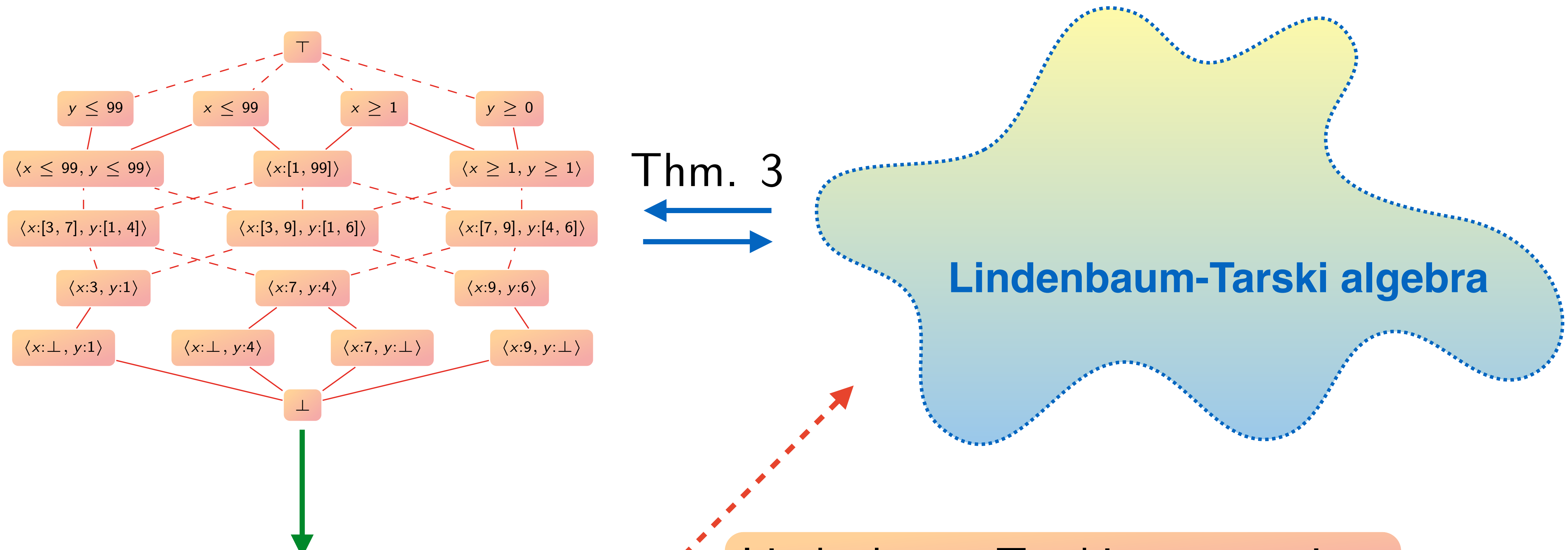
$$\frac{\Gamma, \varphi \vdash \theta}{\Gamma, \varphi \wedge \psi \vdash \theta}\ \wedge\text{L}_1 \qquad \frac{\Gamma, \psi \vdash \theta}{\Gamma, \varphi \wedge \psi \vdash \theta}\ \wedge\text{L}_2 \qquad \frac{\Gamma \vdash \varphi \qquad \Sigma \vdash \psi}{\Gamma, \Sigma \vdash \varphi \wedge \psi}\ \wedge\text{R}$$

$$[m \leq n]\ \frac{\Gamma, x \leq n \vdash \varphi}{\Gamma, x \leq m \vdash \varphi}\ \text{UB-L} \qquad [m \leq n]\ \frac{\Gamma \vdash x \leq m}{\Gamma \vdash x \leq n}\ \text{UB-R}$$

$$[m \leq n]\ \frac{\Gamma, x \geq m \vdash \varphi}{\Gamma, x \geq n \vdash \varphi}\ \text{LB-L} \qquad [m \leq n]\ \frac{\Gamma \vdash x \geq n}{\Gamma \vdash x \geq m}\ \text{LB-R}$$

$$[m < n]\ \frac{}{\Gamma, x \leq m \wedge x \geq n \vdash \mathsf{ff}_x}\ \mathsf{ff}\text{R}_5$$

- how can we prove that the logic captures the lattice?



$\varphi \;::=\; \mathsf{tt} \mid x \leq k \mid x \geq k \mid \varphi \wedge \varphi \mid \mathsf{ff}_x$

Lindenbaum-Tarski construction

$$\frac{}{\Gamma \vdash \mathsf{tt}}\,\text{ttR} \qquad \frac{}{\mathsf{ff}_x \vdash \varphi(x)}\,\text{ffL}$$

$$\frac{}{\varphi \vdash \varphi}\,\text{I} \qquad \frac{\Gamma \vdash \varphi \quad \varphi, \Sigma \vdash \psi}{\Gamma, \Sigma \vdash \psi}\,\text{CUT} \qquad \frac{\Gamma \vdash \psi}{\Gamma, \varphi \vdash \psi}\,\text{WL} \qquad \frac{\Gamma, \varphi, \varphi \vdash \psi}{\Gamma, \varphi \vdash \psi}\,\text{CL} \qquad \frac{\Gamma, \varphi, \psi \vdash \theta}{\Gamma, \psi, \varphi \vdash \theta}\,\text{PL}$$

$$\frac{\Gamma, \varphi \vdash \theta}{\Gamma, \varphi \wedge \psi \vdash \theta}\,\wedge\text{L}_1 \qquad \frac{\Gamma, \psi \vdash \theta}{\Gamma, \varphi \wedge \psi \vdash \theta}\,\wedge\text{L}_2 \qquad \frac{\Gamma \vdash \varphi \quad \Sigma \vdash \psi}{\Gamma, \Sigma \vdash \varphi \wedge \psi}\,\wedge\text{R}$$

$$[m \leq n]\ \frac{\Gamma, x \leq n \vdash \varphi}{\Gamma, x \leq m \vdash \varphi}\,\text{UB-L} \qquad [m \leq n]\ \frac{\Gamma \vdash x \leq m}{\Gamma \vdash x \leq n}\,\text{UB-R}$$

$$[m \leq n]\ \frac{\Gamma, x \geq m \vdash \varphi}{\Gamma, x \geq n \vdash \varphi}\,\text{LB-L} \qquad [m \leq n]\ \frac{\Gamma \vdash x \geq n}{\Gamma \vdash x \geq m}\,\text{LB-R}$$

$$[m < n]\ \frac{}{\Gamma, x \leq m \wedge x \geq n \vdash \mathsf{ff}_x}\,\text{ffR}_5$$

$$\langle \mathcal{I}, \vdash \rangle$$

Lindenbaum-Tarski construction

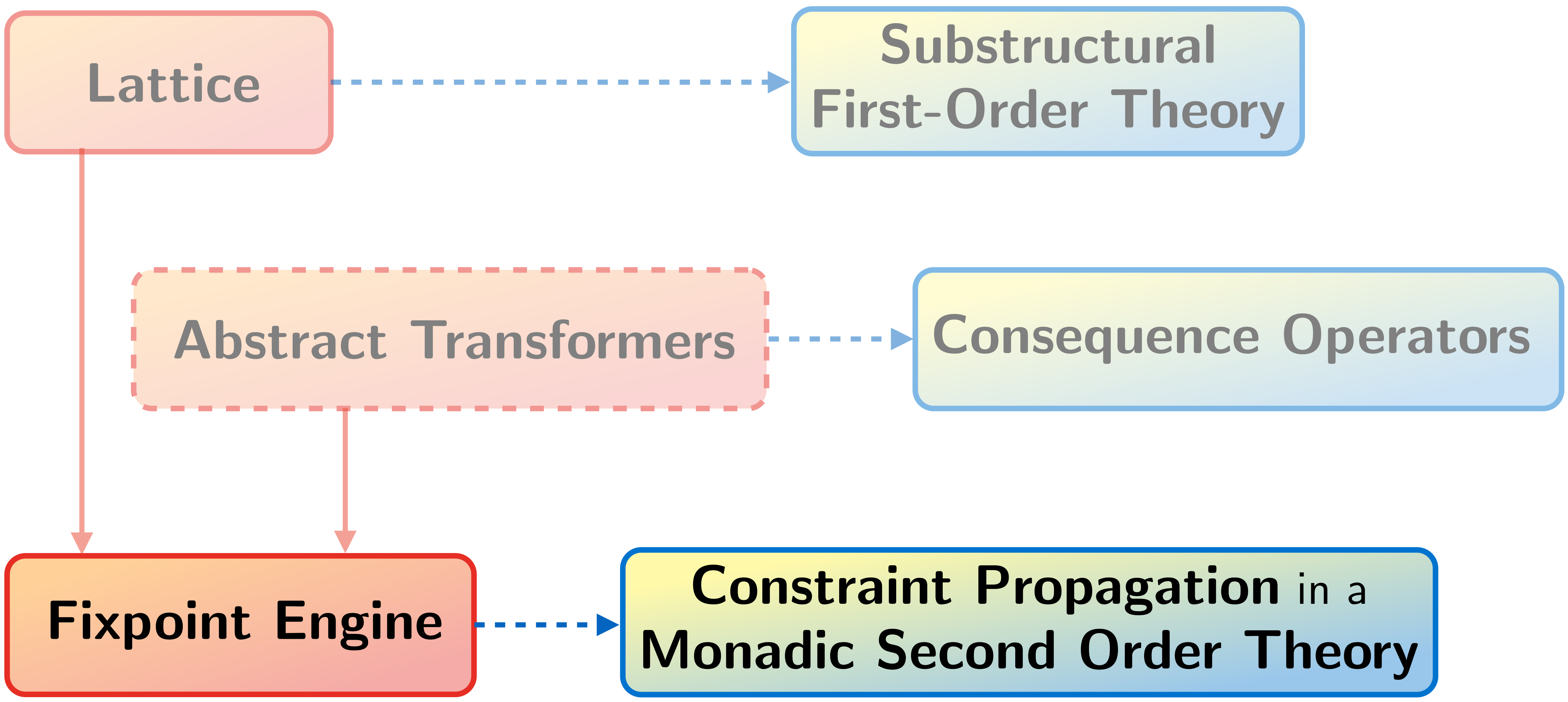
$$\langle \mathcal{I}/\equiv, \sqsubseteq, \sqcap \rangle$$

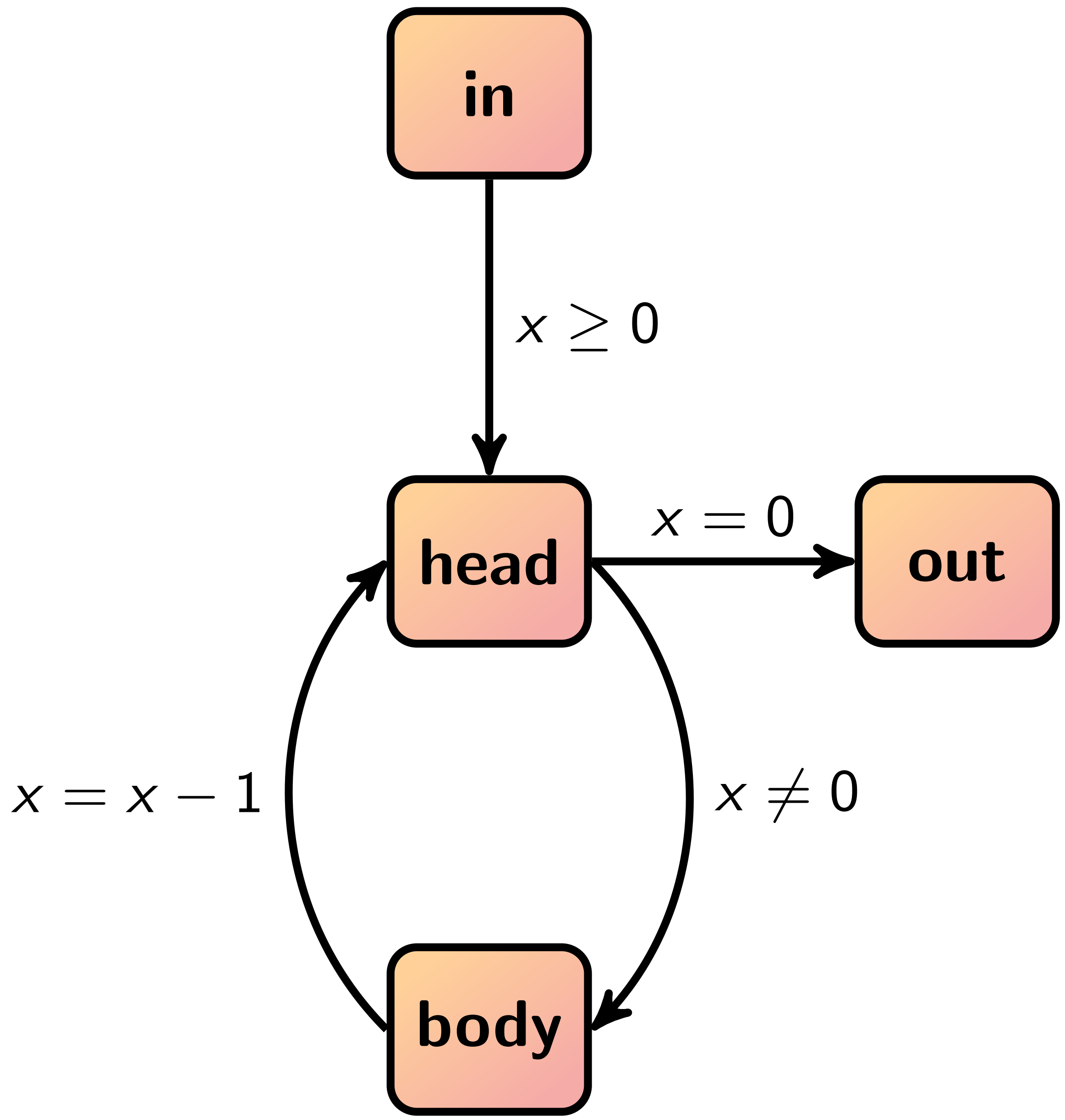
$\varphi \equiv \psi$ if $\varphi \vdash \psi$ and $\psi \vdash \varphi$

$x \leq 5 \equiv x \leq 5 \wedge x \leq 6$

$\varphi \sqsubseteq \psi$ if $\theta_1 \vdash \theta_2$ for $\theta_1 \in [\varphi]$ and $\theta_2 \in [\varphi]$

$\varphi \sqcap \psi$ if $[\theta_1 \wedge \theta_2]$ for $\theta_1 \in [\varphi]$ and $\theta_2 \in [\varphi]$


Lattice
Substructural First-Order Theory
Abstract Transformers
Consequence Operators
Fixpoint Engine
Constraint Propagation in a Monadic Second Order Theory



in	$\mapsto$	$x : (-\infty, +\infty)$
head	$\mapsto$	$x : [0, +\infty)$
body	$\mapsto$	$x : [1, +\infty)$
out	$\mapsto$	$x : [0, 0]$

variable $\mapsto$ constraints

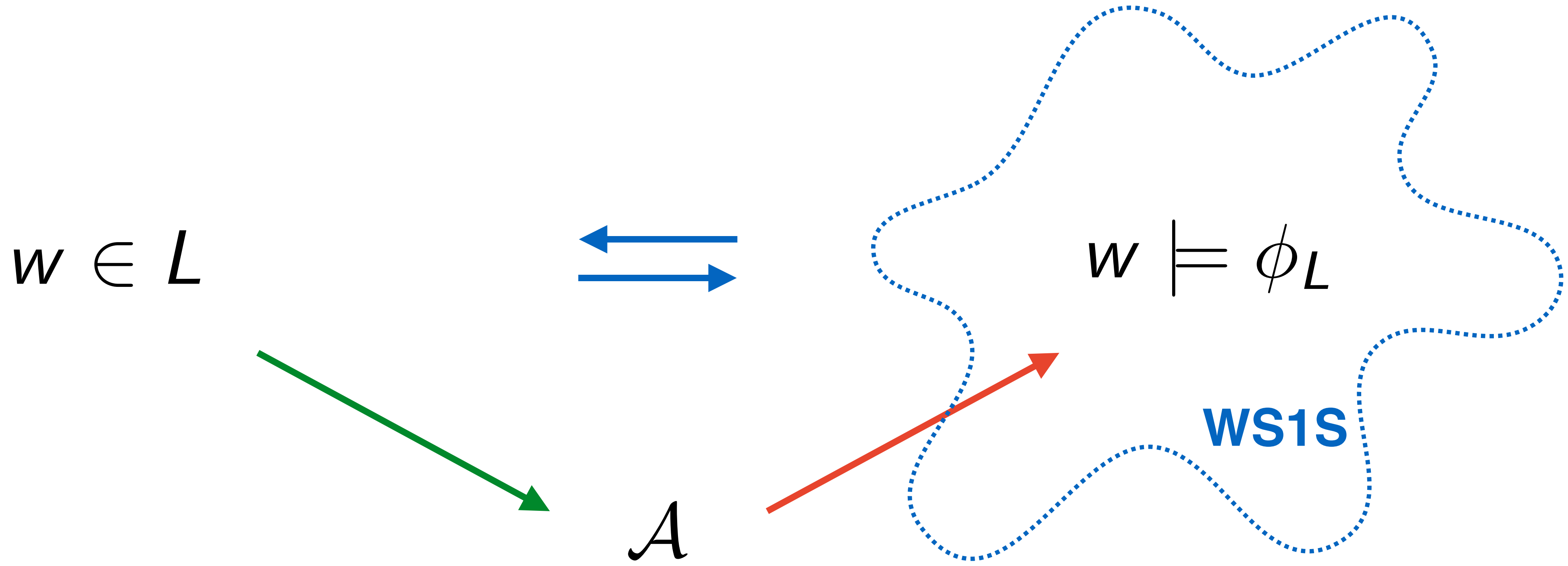
is invariant construction a form of SAT solving?

$$(\mathbf{w} \vee \mathbf{z}) \wedge (\mathbf{y} \vee \mathbf{z}) \wedge (\neg \mathbf{w} \vee \neg \mathbf{z}) \wedge (\neg \mathbf{y} \vee \mathbf{z})$$

w	$\mapsto$	*false*
y	$\mapsto$	*unknown*
z	$\mapsto$	*true*

Büchi's Theorem

a language L is regular if and only if it is expressible in WS1S



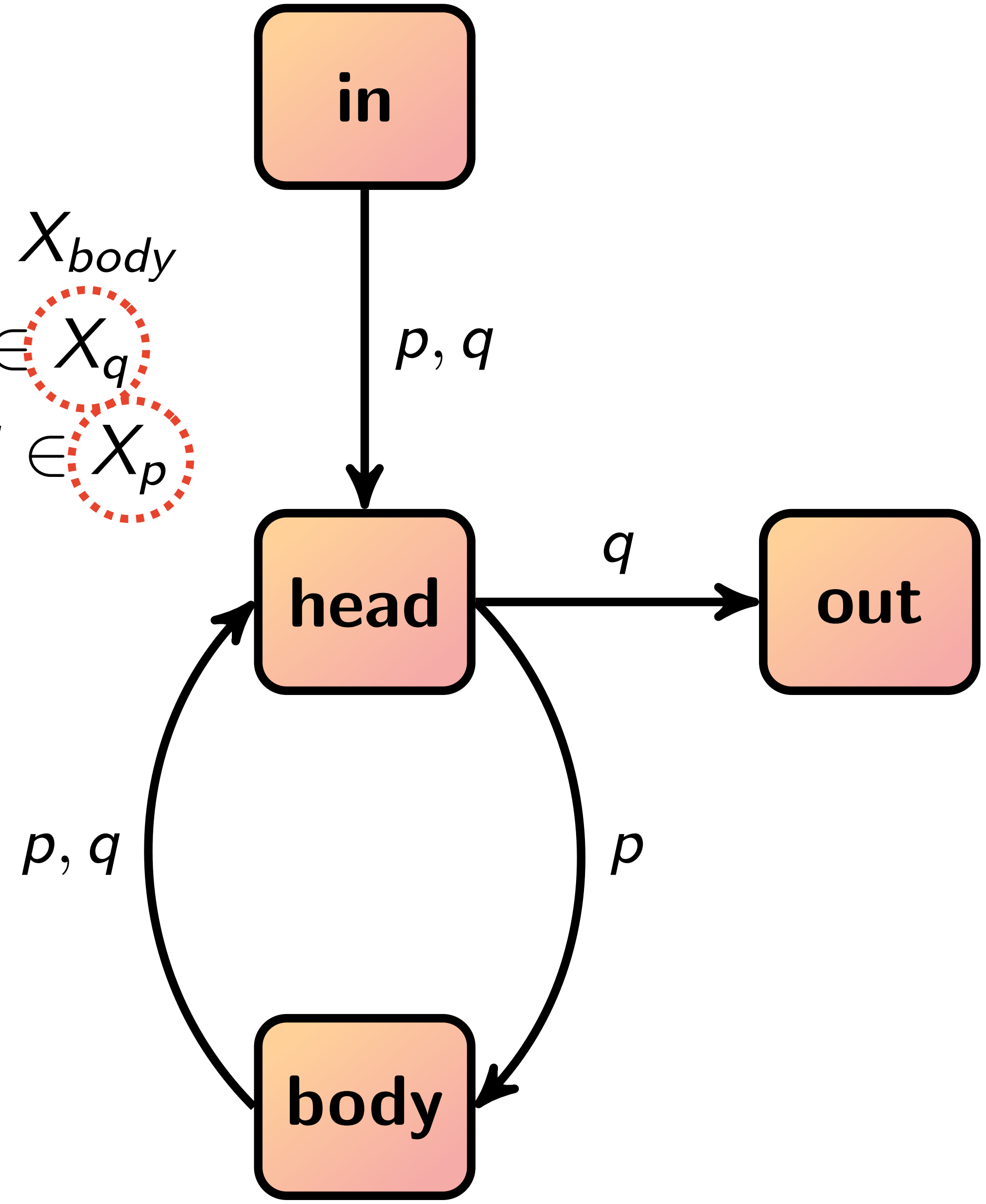
Büchi's Theorem

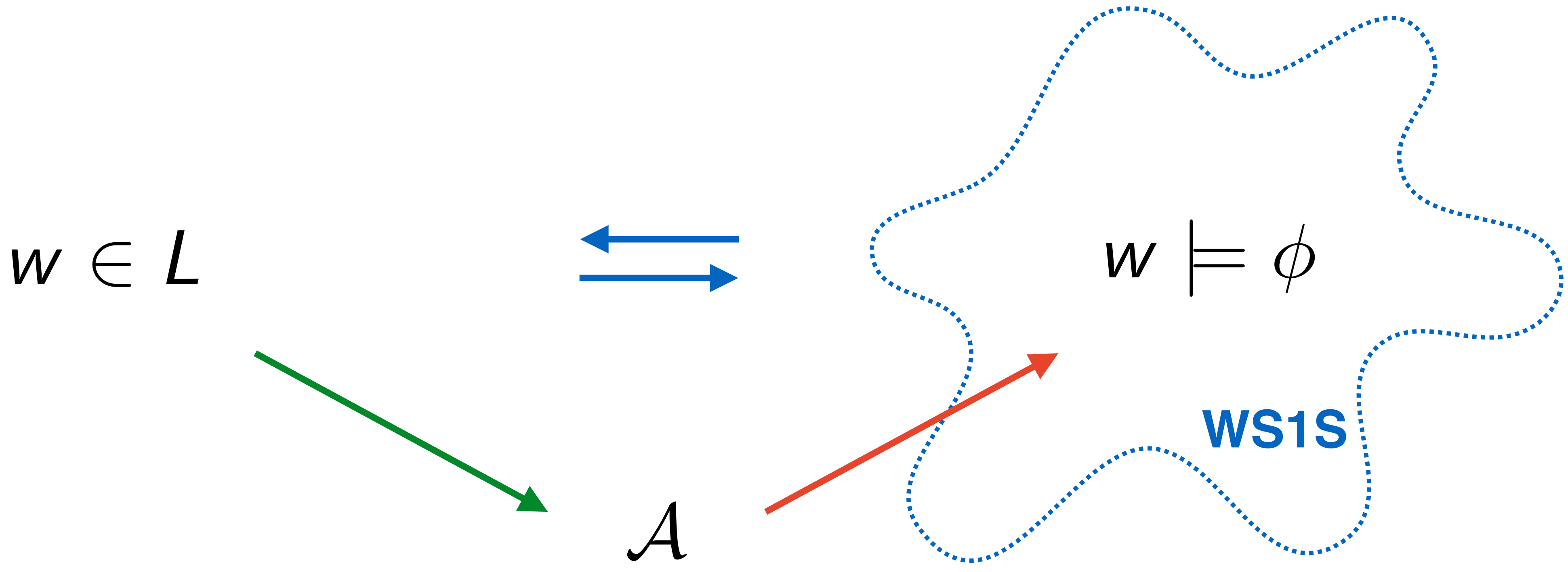
a language L is regular if and only if it is expressible in WS1S

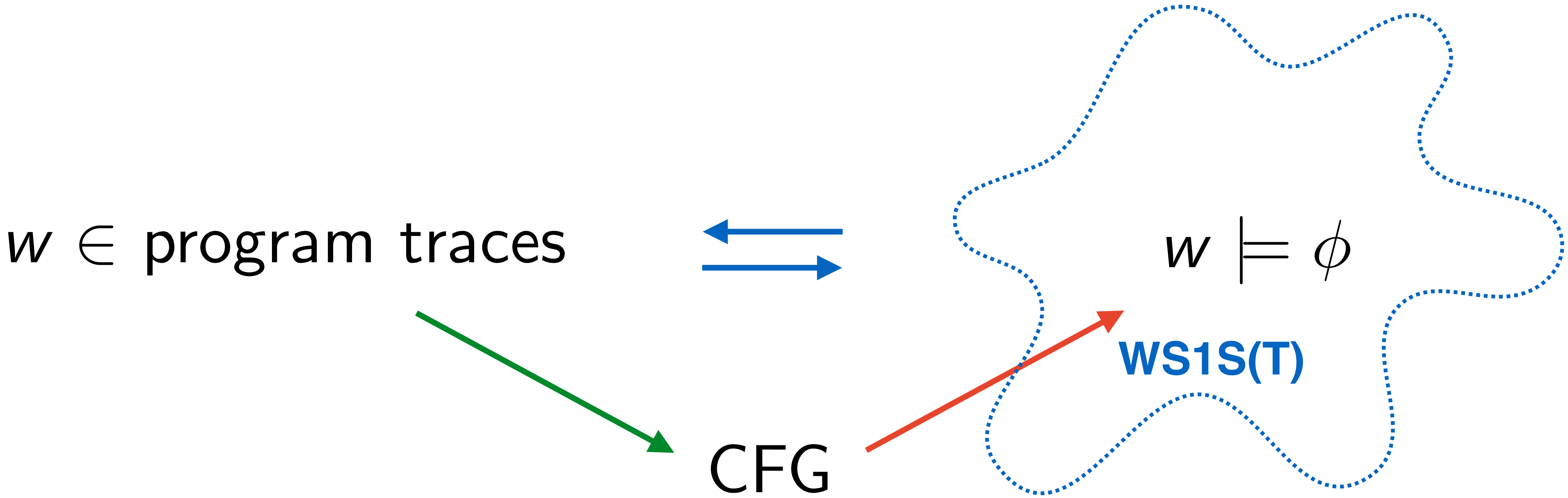
$\forall j : \neg Succ(j, i)$

$$
\begin{aligned}
& \forall i : First(i) \rightarrow i \in X_{in} \\
\wedge \quad & \forall i\, \forall j : j \in X_{head} \wedge Succ(i, j) \rightarrow i \in X_{in} \vee i \in X_{body} \\
\wedge \quad & \forall i\, \forall j : j \in X_{out} \wedge Succ(i, j) \rightarrow i \in X_{head} \wedge i \in X_{q} \\
\wedge \quad & \forall i\, \forall j : j \in X_{body} \wedge Succ(i, j) \rightarrow i \in X_{head} \wedge i \in X_{p} \\
\wedge \quad & \forall i : Last(i) \rightarrow i \in X_{out}
\end{aligned}
$$

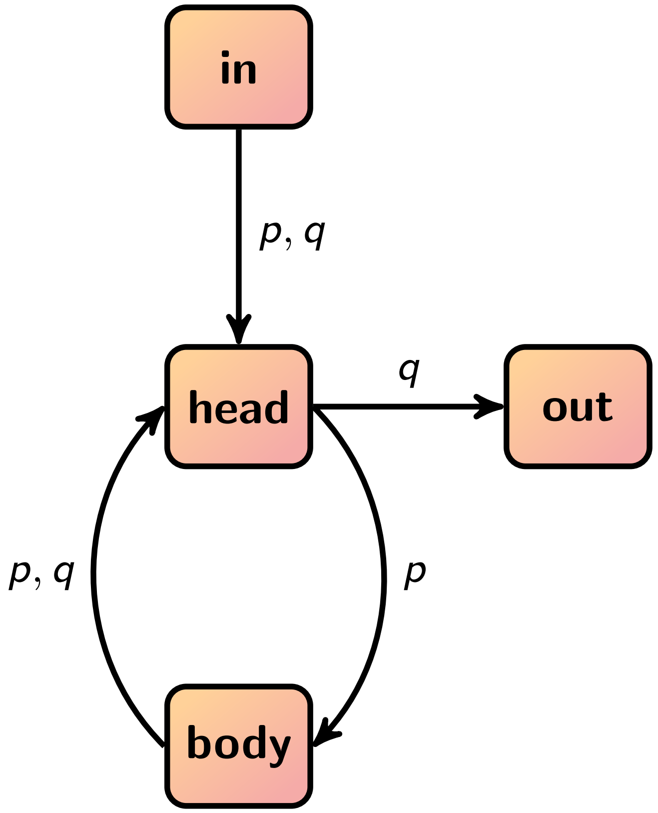
$\forall j : \neg Succ(i, j)$




$w \in L$
$w \models \phi$
WS1S
$\mathcal{A}$


$w \in$ program traces
$w \models \phi$
WS1S(T)
CFG

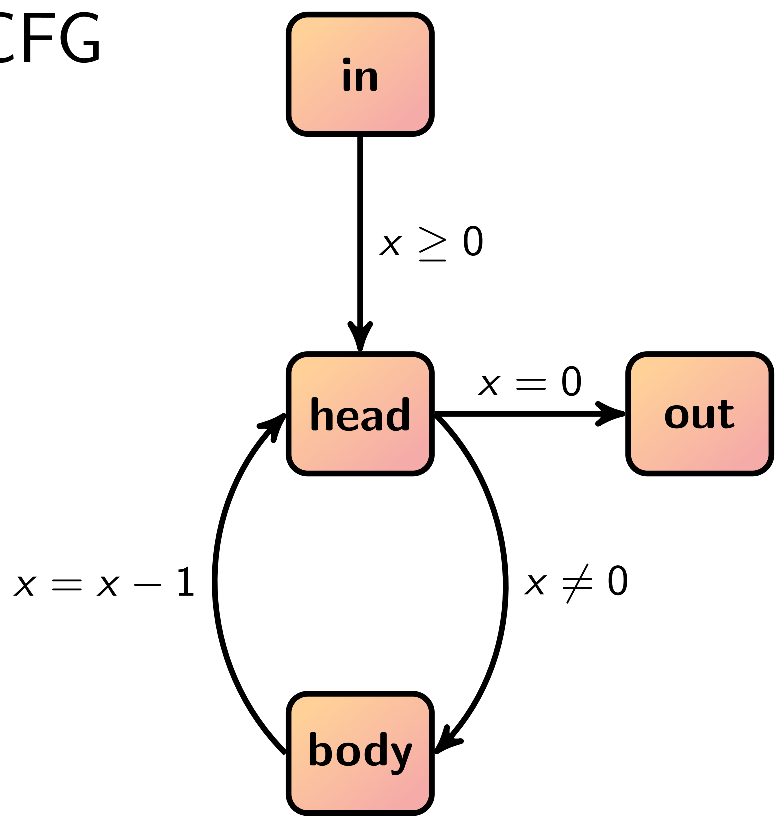
$\mathcal{A}$



WS1S

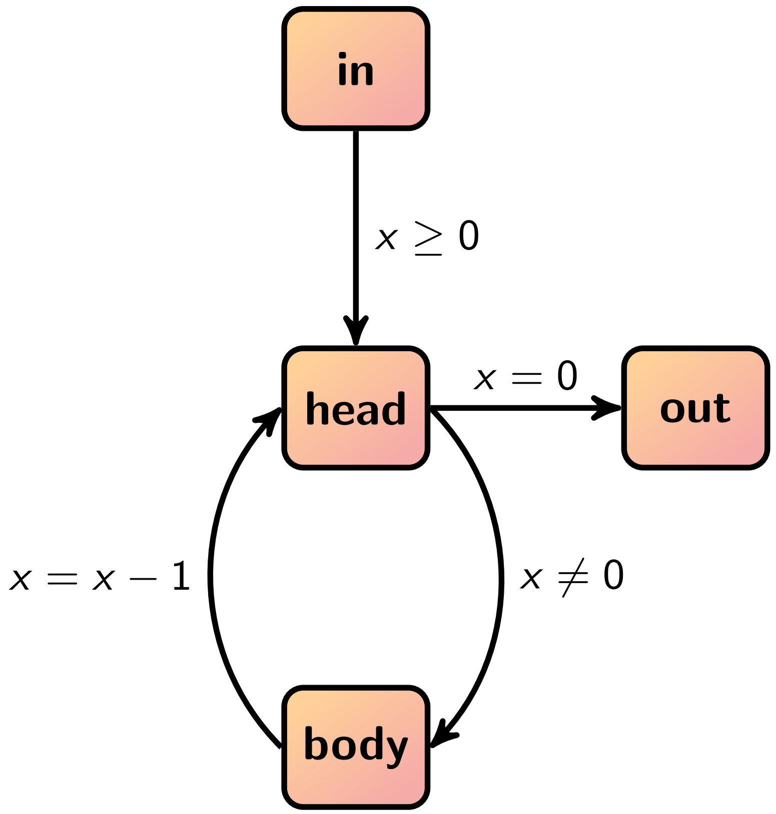
$$
\begin{aligned}
& \forall i : First(i) \rightarrow i \in X_{in} \\
\wedge \quad & \forall i \, \forall j : j \in X_{head} \wedge Succ(i,j) \rightarrow i \in X_{in} \vee i \in X_{body} \\
\wedge \quad & \forall i \, \forall j : j \in X_{out} \wedge Succ(i,j) \rightarrow i \in X_{head} \wedge i \in X_q \\
\wedge \quad & \forall i \, \forall j : j \in X_{body} \wedge Succ(i,j) \rightarrow i \in X_{head} \wedge i \in X_p \\
\wedge \quad & \forall i : Last(i) \rightarrow i \in X_{out}
\end{aligned}
$$

CFG



WS1S(T)

$$
\begin{aligned}
& \forall i : First(i) \rightarrow i \in X_{in} \\
\wedge \quad & \forall i \, \forall j : j \in X_{head} \wedge Succ(i,j) \rightarrow i \in X_{in} \wedge (x \geq 0 \rightarrow succ(x) = x)(i) \\
\wedge \quad & \forall i \, \forall j : j \in X_{out} \wedge Succ(i,j) \rightarrow i \in X_{head} \wedge (x = 0 \rightarrow succ(x) = x)(i) \\
\wedge \quad & \forall i \, \forall j : j \in X_{body} \wedge Succ(i,j) \rightarrow i \in X_{head} \wedge (x \neq 0 \rightarrow succ(x) = x)(i) \\
\wedge \quad & \forall i \, \forall j : j \in X_{head} \wedge Succ(i,j) \rightarrow i \in X_{body} \wedge (succ(x) = x - 1)(i) \\
\wedge \quad & \forall i : Last(i) \rightarrow i \in X_{out}
\end{aligned}
$$



$$
\begin{aligned}
& \forall i : First(i) \rightarrow i \in X_{in} \\
\wedge\ & \forall i\, \forall j : j \in X_{head} \wedge Succ(i,j) \rightarrow i \in X_{in} \wedge (x \geq 0 \rightarrow succ(x) = x)(i) \\
\wedge\ & \forall i\, \forall j : j \in X_{out} \wedge Succ(i,j) \rightarrow i \in X_{head} \wedge (x = 0 \rightarrow succ(x) = x)(i) \\
\wedge\ & \forall i\, \forall j : j \in X_{body} \wedge Succ(i,j) \rightarrow i \in X_{head} \wedge (x \neq 0 \rightarrow succ(x) = x)(i) \\
\wedge\ & \forall i\, \forall j : j \in X_{head} \wedge Succ(i,j) \rightarrow i \in X_{body} \wedge (succ(x) = x - 1)(i) \\
\wedge\ & \forall i : Last(i) \rightarrow i \in X_{out}
\end{aligned}
$$

$$
\begin{aligned}
\textbf{in} &\mapsto x : (-\infty, +\infty) \\
\textbf{head} &\mapsto x : [0, +\infty) \\
\textbf{body} &\mapsto x : [1, +\infty) \\
\textbf{out} &\mapsto x : [0, 0]
\end{aligned}
$$

Theorem

an **abstract interpreter** is a **sound** but *incomplete* **solver** for satisfiability of these formulae

Conflict-Driven Conditional Termination

Vijay D'Silva[1] and Caterina Urban[2]

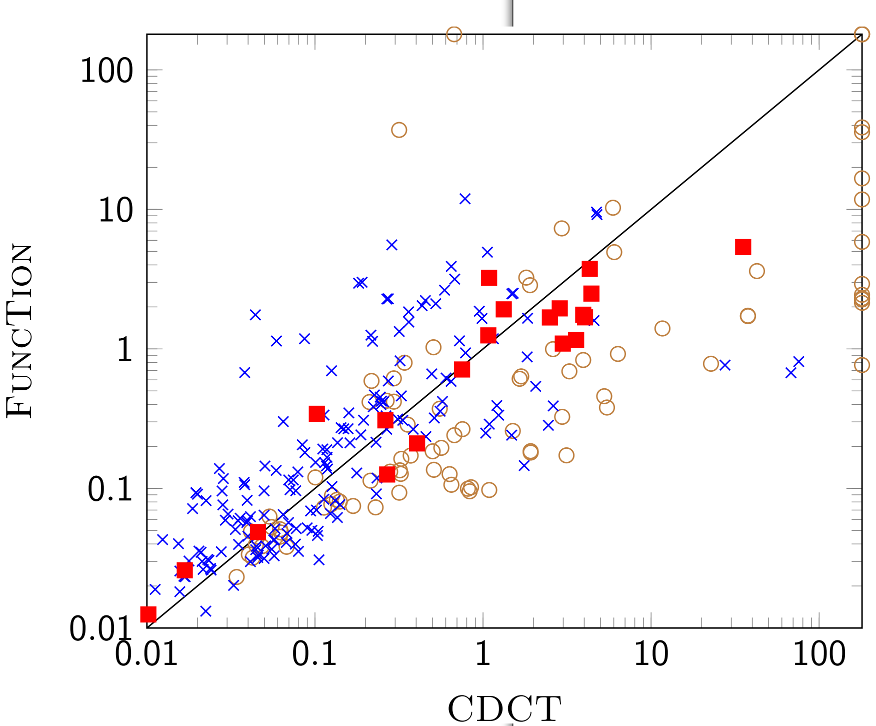
[1] Google Inc., San Francisco
[2] École Normale Supérieure, Paris

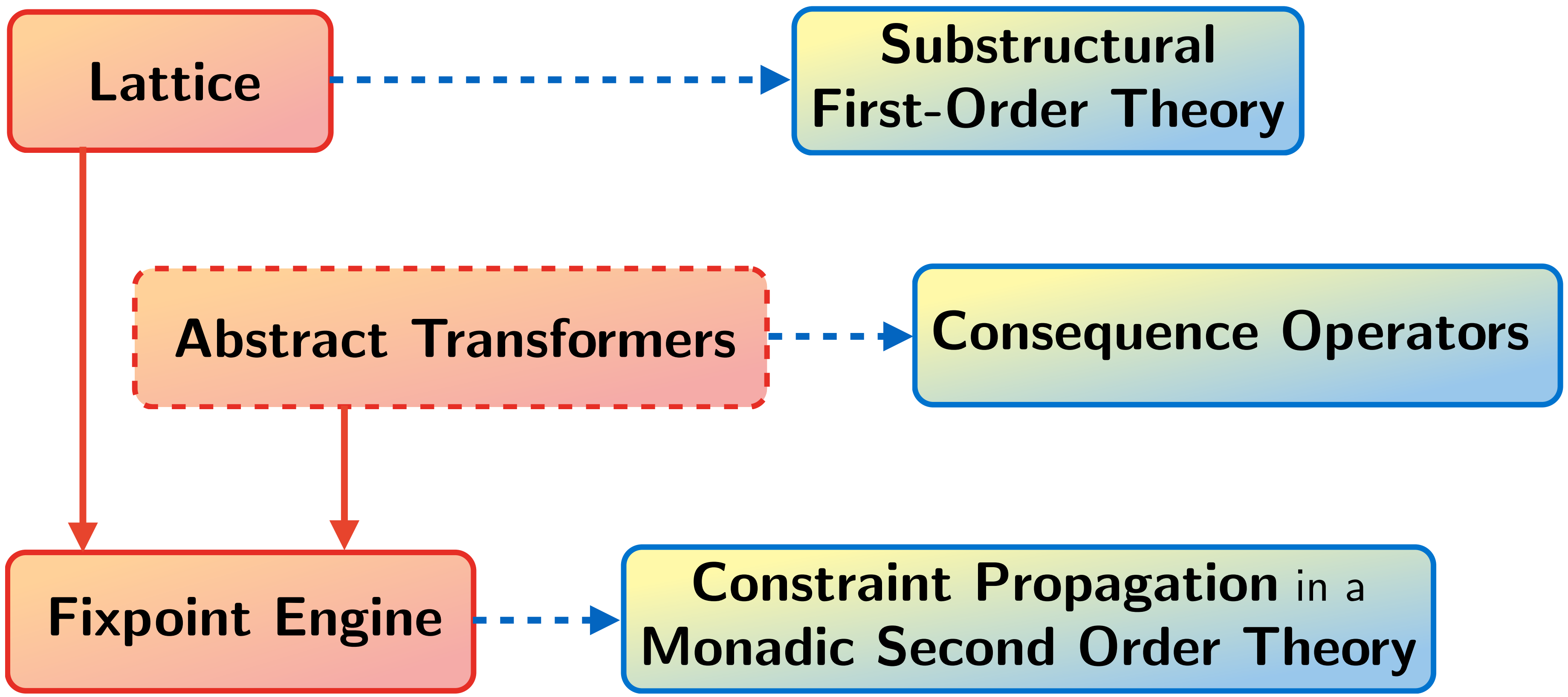
Abstract. Conflict-driven learning, which is essential to the performance of SAT and SMT solvers, consists of a procedure that searches for a model of a formula, and refutation procedure for proving that no model exists. This paper shows that conflict-driven learning can improve the precision of a termination analysis based on abstract interpretation. We encode non-termination as satisfiability in a monadic second-order logic and use abstract interpreters to reason about the satisfiability of this formula. Our search procedure combines decisions with reachability analysis to find potentially non-terminating executions and our refutation procedure uses a conditional termination analysis. Our implementation extends the set of conditional termination arguments discovered by an existing termination analyzer.

1 Conflict-Driven Learning for Termination

Conflict-driven learning procedures are integral to the performance of SAT and SMT solvers. Such procedures combine search and refutation to determine if a formula is satisfiable. Conflicts discovered by search drive refutation, and search learns from refutation to avoid regions of the search space without solutions.

Our work is driven by the observation that discovering a small number of disjunctive termination arguments is crucial to the performance of certain termination analyzers [27]. Fig. 1 summarizes our lifting of conflict-driven learning to termination analysis. We use reachability analysis to find a set of states that constitute potentially non-terminating execution. We apply a conditional termi-





Future Work

- general theory for **non-Cartesian** abstract domains
- integration of decision rules from SAT solvers into static analyzers
- **proof generation** from static analysis