TECHNISCHE
UNIVERSITÄT
DRESDEN

DRESDEN
concept

Institute of Theoretical Computer Science Chair of Automata Theory

THE INCLUSION PROBLEM FOR WEIGHTED AUTOMATA ON INFINITE TREES

Stefan Borgwardt Rafael Peñaloza

Debrecen, August 21, 2011

Introduction

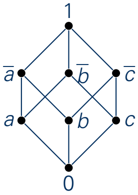
- Automata on infinite trees can recognize tree-shaped models
- Emptiness test useful to decide satisfiability in logics
- Inclusion test could be used to decide entailment
- Here: generalization to lattice-weighted automata

Lattices

De Morgan lattice:

- Bounded distributive lattice $L = (L, \oplus, \otimes, 0, 1)$
- De Morgan negation $^{-} : L \to L$

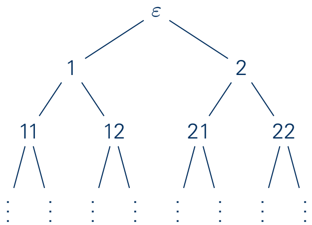
$\overline{x} = 1 - x$



Trees

Infinite k-ary trees:

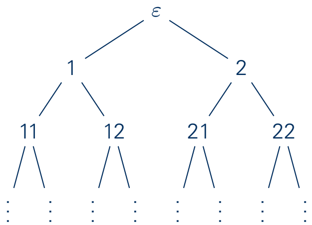
- Nodes are identified by their positions in K^*, where $K := \{1, \ldots, k\}$



Trees

Infinite k-ary trees:

- Nodes are identified by their positions in K^*, where $K := \{1, \ldots, k\}$



- A labeled tree $t \in \Sigma^{K^*}$ is a function $t : K^* \to \Sigma$

Automata

Tree automaton $\mathcal{A} = (Q, \Sigma, I, \Delta, \mathfrak{X})$:

- states Q
- input alphabet Σ
- initial state set $I \subseteq Q$
- transition relation $\Delta \subseteq Q \times \Sigma \times Q^k$
- acceptance condition $\mathfrak{X} \subseteq Q^\omega$

Automata

Tree automaton $\mathcal{A} = (Q, \Sigma, I, \Delta, \mathfrak{X})$:

- states Q
- input alphabet Σ
- initial state set $I \subseteq Q$
- transition relation $\Delta \subseteq Q \times \Sigma \times Q^k$
- acceptance condition $\mathfrak{X} \subseteq Q^\omega$

Successful run $r \in \text{succ}(\mathcal{A}) \subseteq Q^{K^*}$: every path is in $\mathfrak{X}$

Automata

Tree automaton $\mathcal{A} = (Q, \Sigma, I, \Delta, \mathfrak{X})$:

- states Q
- input alphabet Σ
- initial state set $I \subseteq Q$
- transition relation $\Delta \subseteq Q \times \Sigma \times Q^k$
- acceptance condition $\mathfrak{X} \subseteq Q^\omega$

Successful run $r \in \mathrm{succ}(\mathcal{A}) \subseteq Q^{K^*}$: every path is in $\mathfrak{X}$

Transition of r on $t \in \Sigma^{K^*}$ at $u \in K^*$: $\overrightarrow{r(t,u)} := (r(u), t(u), r(u1), \ldots, r(uk))$

Automata

Tree automaton $\mathcal{A} = (Q, \Sigma, I, \Delta, \mathfrak{X})$:

- states Q
- input alphabet Σ
- initial state set $I \subseteq Q$
- transition relation $\Delta \subseteq Q \times \Sigma \times Q^k$
- acceptance condition $\mathfrak{X} \subseteq Q^\omega$

Successful run $r \in \text{succ}(\mathcal{A}) \subseteq Q^{K^*}$: every path is in $\mathfrak{X}$

Transition of r on $t \in \Sigma^{K^*}$ at $u \in K^*$: $\overrightarrow{r(t, u)} := (r(u), t(u), r(u1), \ldots, r(uk))$

$$t \in L(\mathcal{A}) \text{ iff } \exists_{r \in \text{succ}(\mathcal{A})} r(\varepsilon) \in I \wedge \forall_{u \in K^*} \overrightarrow{r(t, u)} \in \Delta$$

Automata

Weighted tree automaton $\mathcal{A} = (Q, \Sigma, S, \text{in}, \text{wt}, \mathfrak{X})$:

- states Q
- input alphabet Σ
- (finite) distributive lattice S
- initial distribution $\text{in} : Q \to S$
- transition weight function $\text{wt} : Q \times \Sigma \times Q^k \to S$
- acceptance condition $\mathfrak{X} \subseteq Q^\omega$

Successful run $r \in \text{succ}(\mathcal{A}) \subseteq Q^{K^*}$: every path is in $\mathfrak{X}$

Transition of r on $t \in \Sigma^{K^*}$ at $u \in K^*$: $\overrightarrow{r(t,u)} := (r(u), t(u), r(u1), \ldots, r(uk))$

$$t \in L(\mathcal{A}) \text{ iff } \exists_{r \in \text{succ}(\mathcal{A})} r(\varepsilon) \in I \wedge \forall_{u \in K^*} \overrightarrow{r(t,u)} \in \Delta$$

Automata

Weighted tree automaton $\mathcal{A} = (Q, \Sigma, S, \mathrm{in}, \mathrm{wt}, \mathfrak{X})$:

- states Q
- input alphabet Σ
- (finite) distributive lattice S
- initial distribution $\mathrm{in} : Q \to S$
- transition weight function $\mathrm{wt} : Q \times \Sigma \times Q^k \to S$
- acceptance condition $\mathfrak{X} \subseteq Q^\omega$

Successful run $r \in \mathrm{succ}(\mathcal{A}) \subseteq Q^{K^*}$: every path is in $\mathfrak{X}$

Transition of r on $t \in \Sigma^{K^*}$ at $u \in K^*$: $\overrightarrow{r(t,u)} := (r(u), t(u), r(u1), \ldots, r(uk))$

$$(\|\mathcal{A}\|, t) = \bigoplus_{r \in \mathrm{succ}(\mathcal{A})} \mathrm{in}(r(\varepsilon)) \otimes \bigotimes_{u \in K^*} \mathrm{wt}(\overrightarrow{r(t,u)})$$

Automata

Weighted tree automaton $\mathcal{A} = (Q, \Sigma, S, \text{in}, \text{wt}, \mathfrak{X})$:

- states Q
- input alphabet Σ
- (finite) distributive lattice S
- initial distribution $\text{in} : Q \to S$
- transition weight function $\text{wt} : Q \times \Sigma \times Q^k \to S$
- acceptance condition $\mathfrak{X} \subseteq Q^\omega$

Successful run $r \in \text{succ}(\mathcal{A}) \subseteq Q^{K^*}$: every path is in $\mathfrak{X}$

Transition of r on $t \in \Sigma^{K^*}$ at $u \in K^*$: $\overrightarrow{r(t,u)} := (r(u), t(u), r(u1), \ldots, r(uk))$

$$(\|\mathcal{A}\|, t) = \bigoplus_{r \in \text{succ}(\mathcal{A})} \text{in}(r(\varepsilon)) \otimes \bigotimes_{u \in K^*} \text{wt}(\overrightarrow{r(t,u)})$$

Acceptance conditions: looping, Büchi, co-Büchi, parity

Automata (2)

PTIME-problems for Büchi automata:

- Infimum of two automata: $(\|\mathcal{C}\|, t) = (\|\mathcal{A}\|, t) \otimes (\|\mathcal{B}\|, t)$
- Supremum of two automata: $(\|\mathcal{C}\|, t) = (\|\mathcal{A}\|, t) \oplus (\|\mathcal{B}\|, t)$
- Computing the behavior $\bigoplus_{t \in \Sigma^{K*}} (\|\mathcal{A}\|, t)$ [Baader, Peñaloza 2010]

Description Logics

The description logic $\mathcal{ALC}$	Syntax
concept name	$A \in N_C$
role name	$r \in N_R$

Description Logics

The description logic $\mathcal{ALC}$	Syntax	interpretation $\mathcal{I} = (\cdot^{\mathcal{I}}, \Delta^{\mathcal{I}})$
concept name	$A \in N_C$	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
role name	$r \in N_R$	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Description Logics

The description logic $\mathcal{ALC}$	Syntax	interpretation $\mathcal{I} = (\cdot^\mathcal{I}, \Delta^\mathcal{I})$
concept name	$A \in N_C$	$A^\mathcal{I} \subseteq \Delta^\mathcal{I}$
role name	$r \in N_R$	$r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$
top concept	$\top$	$\Delta^\mathcal{I}$
bottom concept	$\bot$	$\emptyset$
conjunction	$C \sqcap D$	$C^\mathcal{I} \cap D^\mathcal{I}$
disjunction	$C \sqcup D$	$C^\mathcal{I} \cup D^\mathcal{I}$
negation	$\neg C$	$\overline{C^\mathcal{I}}$

Description Logics

The description logic $\mathcal{ALC}$	Syntax	interpretation $\mathcal{I} = (\cdot^{\mathcal{I}}, \Delta^{\mathcal{I}})$
concept name	$A \in N_C$	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
role name	$r \in N_R$	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
top concept	$\top$	$\Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\emptyset$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
negation	$\neg C$	$\overline{C^{\mathcal{I}}}$
existential restriction	$\exists r.C$	$\{x \mid \exists y : (x, y) \in r^{\mathcal{I}} \land y \in C^{\mathcal{I}}\}$
universal restriction	$\forall r.C$	$\{x \mid \forall y : (x, y) \in r^{\mathcal{I}} \to y \in C^{\mathcal{I}}\}$

Description Logics

The description logic $\mathcal{ALC}$	Syntax	interpretation $\mathcal{I} = (\cdot^{\mathcal{I}}, \Delta^{\mathcal{I}})$
concept name	$A \in N_C$	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
role name	$r \in N_R$	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
top concept	$\top$	$\Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\emptyset$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
negation	$\neg C$	$\overline{C^{\mathcal{I}}}$
existential restriction	$\exists r.C$	$\{x \mid \exists y : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
universal restriction	$\forall r.C$	$\{x \mid \forall y : (x, y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$
terminological axiom	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$

Description Logics

The description logic $\mathcal{ALC}$	Syntax	interpretation $\mathcal{I} = (\cdot^{\mathcal{I}}, \Delta^{\mathcal{I}})$
concept name	$A \in N_C$	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
role name	$r \in N_R$	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
top concept	$\top$	$\Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\emptyset$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
negation	$\neg C$	$\overline{C^{\mathcal{I}}}$
existential restriction	$\exists r.C$	$\{x \mid \exists y : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
universal restriction	$\forall r.C$	$\{x \mid \forall y : (x, y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$
terminological axiom	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$

- Consistency of a TBox $\mathcal{T}$ (set of axioms): Is there a model of $\mathcal{T}$?
- Satisfiability of C w.r.t. $\mathcal{T}$: Is there a model $\mathcal{I}$ of $\mathcal{T}$ with $C^{\mathcal{I}} \neq \emptyset$?
- Subsumption $C \sqsubseteq_{\mathcal{T}} D$: Does $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ hold in all models $\mathcal{I}$ of $\mathcal{T}$?

Tree Models

- $\mathcal{ALC}$ has the tree model property
- Satisfiability of C w.r.t. $\mathcal{T}$ can be reduced to emptiness of $\mathcal{A}_{C,\mathcal{T}}$

Tree Models

- $\mathcal{ALC}$ has the tree model property
- Satisfiability of C w.r.t. $\mathcal{T}$ can be reduced to emptiness of $\mathcal{A}_{C,\mathcal{T}}$
- Behavior computation can be used for axiom pinpointing

  ↳ identifying the axioms of $\mathcal{T}$ that are responsible for a contradiction

Tree Models

- $\mathcal{ALC}$ has the tree model property
- Satisfiability of C w.r.t. $\mathcal{T}$ can be reduced to emptiness of $\mathcal{A}_{C,\mathcal{T}}$
- Behavior computation can be used for axiom pinpointing

  ↳ identifying the axioms of $\mathcal{T}$ that are responsible for a contradiction
- $C \sqsubseteq_{\mathcal{T}} D$ iff $C \sqcap \neg D$ is unsatisfiable

Tree Models

- $\mathcal{ALC}$ has the tree model property
- Satisfiability of C w.r.t. $\mathcal{T}$ can be reduced to emptiness of $\mathcal{A}_{C,\mathcal{T}}$
- Behavior computation can be used for axiom pinpointing

  ↳ identifying the axioms of $\mathcal{T}$ that are responsible for a contradiction
- $C \sqsubseteq_{\mathcal{T}} D$ iff $C \sqcap \neg D$ is unsatisfiable
- Inclusion test is useful for non-standard inferences

Inclusion and Complementation

Given two automata $\mathcal{A}$, $\mathcal{A}'$,
does $L(\mathcal{A}') \subseteq L(\mathcal{A})$ hold?

Given an automaton $\mathcal{A}$,
construct an automaton $\overline{\mathcal{A}}$ with $L(\overline{\mathcal{A}}) = \overline{L(\mathcal{A})}$.

Inclusion and Complementation

Given two automata $\mathcal{A}$, $\mathcal{A}'$,

does $L(\mathcal{A}') \subseteq L(\mathcal{A})$ hold?

Given an automaton $\mathcal{A}$,

construct an automaton $\overline{\mathcal{A}}$ with $L(\overline{\mathcal{A}}) = \overline{L(\mathcal{A})}$.

Given two weighted automata $\mathcal{A}$, $\mathcal{A}'$,

compute $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$.

Inclusion and Complementation

Given two automata $\mathcal{A}$, $\mathcal{A}'$,

does $L(\mathcal{A}') \subseteq L(\mathcal{A})$ hold?

Given an automaton $\mathcal{A}$,

construct an automaton $\overline{\mathcal{A}}$ with $L(\overline{\mathcal{A}}) = \overline{L(\mathcal{A})}$.

Given two weighted automata $\mathcal{A}$, $\mathcal{A}'$,

compute $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$.

Given a weighted automaton $\mathcal{A}$,

construct a weighted automaton $\overline{\mathcal{A}}$ with $\|\overline{\mathcal{A}}\| = \overline{\|\mathcal{A}\|}$.

Inclusion and Complementation

Given two automata $\mathcal{A}$, $\mathcal{A}'$,
does $L(\mathcal{A}') \subseteq L(\mathcal{A})$ hold?

Given an automaton $\mathcal{A}$,
construct an automaton $\overline{\mathcal{A}}$ with $L(\overline{\mathcal{A}}) = \overline{L(\mathcal{A})}$.

Given two weighted automata $\mathcal{A}$, $\mathcal{A}'$,
compute $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$.

Given a weighted automaton $\mathcal{A}$,
construct a weighted automaton $\overline{\mathcal{A}}$ with $\|\overline{\mathcal{A}}\| = \overline{\|\mathcal{A}\|}$.

[Buhrke, Lescow, Vöge 1996; Kupferman, Vardi 1998; Vardi, Wilke 2008]:
Inclusion is in EXPTIME for parity automata

[Seidl 1989]:
Inclusion is EXPTIME-hard for automata on finite trees

Glass-box Approach

Modify algorithms for unweighted complementation for weighted automata:

Glass-box Approach

Modify algorithms for unweighted complementation for weighted automata:

- [Miyano, Hayashi 1984; Muller, Schupp 1987]:
  - Exponential constructions for complementing looping and co-Büchi into Büchi automata (powerset construction $Q \rightsquigarrow 2^Q$)

Glass-box Approach

Modify algorithms for unweighted complementation for weighted automata:

- [Miyano, Hayashi 1984; Muller, Schupp 1987]:
  - Exponential constructions for complementing looping and co-Büchi into Büchi automata (powerset construction $Q \rightsquigarrow 2^Q$)

- Translation of the constructions and proofs to finite De Morgan lattices:
  - from 2^Q to S^Q
  - from $\wedge$ to $\otimes$ and $\vee$ to $\oplus$
  - from $\forall$ to $\bigotimes$ and $\exists$ to $\bigoplus$
  - from $q \in I$ to $\mathrm{in}(q)$ and $(\dots) \in \Delta$ to $\mathrm{wt}(\dots)$
  - from $x \Rightarrow y$ to $\overline{x} \oplus y$ or $x \leq y$

Glass-box Approach

Example:

"If r is a successful run of $\mathcal{A}$ on t and r_c is a successul run of $\overline{\mathcal{A}}$ on t, then all paths p of length m have a node $u \in p$ such that $r(u) \notin r_c(u)$."

Glass-box Approach

Example:

"If r is a successful run of $\mathcal{A}$ on t and r_c is a successul run of $\overline{\mathcal{A}}$ on t, then all paths p of length m have a node $u \in p$ such that $r(u) \notin r_c(u)$."

$r \in \text{succ}(\mathcal{A}),\ r_c \in \text{succ}(\overline{\mathcal{A}})$:

$$r \in \Delta(t) \wedge r_c \in \Delta_c(t) \Rightarrow \bigforall_{p \in Path(K*,m)} \ \bigexists_{u \in p} r(u) \notin r_c(u)$$

Glass-box Approach

Example:

"If r is a successful run of $\mathcal{A}$ on t and r_c is a successul run of $\overline{\mathcal{A}}$ on t, then all paths p of length m have a node $u \in p$ such that $r(u) \notin r_c(u)$."

$r \in \text{succ}(\mathcal{A}),\ r_c \in \text{succ}(\overline{\mathcal{A}})$:

$$r \in \Delta(t) \wedge r_c \in \Delta_c(t) \Rightarrow \bigforall_{p \in Path(K*,m)} \bigexists_{u \in p} r(u) \notin r_c(u)$$

$r \in \text{succ}(\mathcal{A}),\ r_c \in \text{succ}(\overline{\mathcal{A}})$:

$$\text{wt}(t,r) \otimes \text{wt}_c(t,r_c) \leq \bigotimes_{p \in Path(K*,m)} \bigoplus_{u \in p} \overline{r_c(u)(r(u))}$$

Glass-box Approach

Example:

"If r is a successful run of $\mathcal{A}$ on t and r_c is a successul run of $\overline{\mathcal{A}}$ on t, then all paths p of length m have a node $u \in p$ such that $r(u) \notin r_c(u)$."

$r \in \text{succ}(\mathcal{A}),\ r_c \in \text{succ}(\overline{\mathcal{A}})$:

$$r \in \Delta(t) \wedge r_c \in \Delta_c(t) \Rightarrow \bigforall_{p \in Path(K*,m)} \bigexists_{u \in p} r(u) \notin r_c(u)$$

$r \in \text{succ}(\mathcal{A}),\ r_c \in \text{succ}(\overline{\mathcal{A}})$:

$$\text{wt}(t, r) \otimes \text{wt}_c(t, r_c) \leq \bigotimes_{p \in Path(K*,m)} \bigoplus_{u \in p} \overline{r_c(u)(r(u))}$$

Only correct for Boolean lattices

Black-box Approach

$p \in S$ meet prime: $a \otimes b \leq p$ implies $a \leq p$ or $b \leq p$

Every $x \in S$ is equal to the infimum of all meet prime elements above x.

Black-box Approach

$p \in S$ meet prime: $a \otimes b \leq p$ implies $a \leq p$ or $b \leq p$

Every $x \in S$ is equal to the infimum of all meet prime elements above x.

Which meet prime elements of S are above $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$?

Black-box Approach

$p \in S$ meet prime: $a \otimes b \leq p$ implies $a \leq p$ or $b \leq p$

Every $x \in S$ is equal to the infimum of all meet prime elements above x.

Which meet prime elements of S are above $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$?

$$\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t) \leq p$$

Black-box Approach

$p \in S$ meet prime: $a \otimes b \leq p$ implies $a \leq p$ or $b \leq p$

Every $x \in S$ is equal to the infimum of all meet prime elements above x.

Which meet prime elements of S are above $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$?

$$\begin{aligned} & \bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t) \leq p \\ \text{iff} \quad & \exists t \in \Sigma^{K^*} : (\|\mathcal{A}'\|, t) \geq \overline{p} \text{ and } (\|\mathcal{A}\|, t) \leq p \end{aligned}$$

Black-box Approach

$p \in S$ meet prime: $a \otimes b \leq p$ implies $a \leq p$ or $b \leq p$

Every $x \in S$ is equal to the infimum of all meet prime elements above x.

Which meet prime elements of S are above $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$?

$$\begin{aligned} & \bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t) \leq p \\ \text{iff} \quad & \exists t \in \Sigma^{K^*} : (\|\mathcal{A}'\|, t) \geq \overline{p} \text{ and } (\|\mathcal{A}\|, t) \leq p \\ \text{iff} \quad & \exists t \in \Sigma^{K^*} : t \in L(\mathcal{A}'_{\geq \overline{p}}) \text{ and } t \notin L(\mathcal{A}_{\not\leq p}) \end{aligned}$$

Black-box Approach

$p \in S$ meet prime: $a \otimes b \leq p$ implies $a \leq p$ or $b \leq p$

Every $x \in S$ is equal to the infimum of all meet prime elements above x.

Which meet prime elements of S are above $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$?

$$
\begin{aligned}
& \bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t) \leq p \\
\text{iff} \quad & \exists t \in \Sigma^{K^*} : (\|\mathcal{A}'\|, t) \geq \overline{p} \text{ and } (\|\mathcal{A}\|, t) \leq p \\
\text{iff} \quad & \exists t \in \Sigma^{K^*} : t \in L(\mathcal{A}'_{\geq \overline{p}}) \text{ and } t \notin L(\mathcal{A}_{\not\leq p}) \\
\text{iff} \quad & L(\mathcal{A}'_{\geq \overline{p}}) \not\subseteq L(\mathcal{A}_{\not\leq p})
\end{aligned}
$$

Black-box Approach

$p \in S$ meet prime: $a \otimes b \leq p$ implies $a \leq p$ or $b \leq p$

Every $x \in S$ is equal to the infimum of all meet prime elements above x.

Which meet prime elements of S are above $\bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t)$?

$$
\begin{aligned}
& \bigotimes_{t \in \Sigma^{K*}} \overline{(\|\mathcal{A}'\|, t)} \oplus (\|\mathcal{A}\|, t) \leq p \\
\text{iff} \quad & \exists t \in \Sigma^{K*} : (\|\mathcal{A}'\|, t) \geq \overline{p} \text{ and } (\|\mathcal{A}\|, t) \leq p \\
\text{iff} \quad & \exists t \in \Sigma^{K*} : t \in L(\mathcal{A}'_{\geq \overline{p}}) \text{ and } t \notin L(\mathcal{A}_{\nleq p}) \\
\text{iff} \quad & L(\mathcal{A}'_{\geq \overline{p}}) \nsubseteq L(\mathcal{A}_{\nleq p})
\end{aligned}
$$

We need exponentially many inclusion tests between unweighted automata.

Conclusions

Summary:

- black-box ($n2^m$) is faster and more general than this glass-box approach (2^{nm})
- optimizations of glass-box algorithm?

Conclusions

Summary:

- black-box ($n2^m$) is faster and more general than this glass-box approach (2^{nm})
- optimizations of glass-box algorithm?

Applications:

- lattice-weighted automata for axiom pinpointing
- automata-based reasoning in fuzzy description logics

Thank You

- Franz Baader and Rafael Peñaloza: Automata-based axiom pinpointing.
  J. Autom. Reasoning, 45(2):91–129, 2010. Special Issue: IJCAR'08.
- Stefan Borgwardt and Rafael Peñaloza: Complementation and inclusion of weighted automata on infinite trees: Revised version.
  LTCS-Report 11-02, Technische Universität Dresden, 2011.
  See http://lat.inf.tu-dresden.de/research/reports.html.
- Stefan Borgwardt and Rafael Peñaloza: Description logics over lattices with multi-valued ontologies.
  In *Proc. IJCAI'11*, pages 768–773. AAAI Press, 2011.
- Orna Kupferman and Yoad Lustig: Lattice automata.
  In *Proc. VMCAI'07*, volume 4349 of *LNCS*, pages 199–213. Springer, 2007.
- Satoru Miyano and Takeshi Hayashi: Alternating finite automata on omega-words.
  Theor. Comput. Sci., 32:321–330, 1984.
- David E. Muller and Paul E. Schupp: Alternating automata on infinite trees.
  Theor. Comput. Sci., 54(2-3):267–276, 1987.