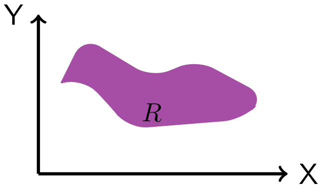
On uniformisability in monadic second-order logic

Michał Skrzypczak

LIAFA, University of Warsaw

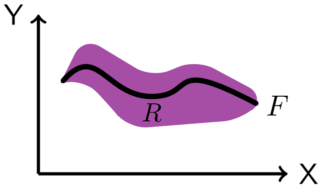
Uniformisation

Uniformisation



Relation $R \subseteq X \times Y$

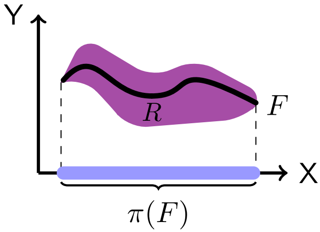
Uniformisation



Relation $R \subseteq X \times Y$

Uniformisation $F \subseteq R$

Uniformisation

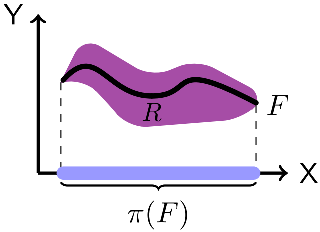


Relation $R \subseteq X \times Y$

Uniformisation $F \subseteq R$

Projection $\pi(F) = \pi(R)$

Uniformisation



Relation $R \subseteq X \times Y$

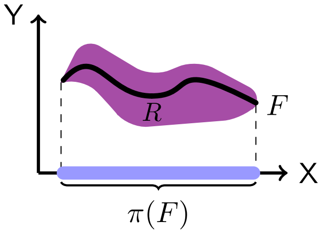
Uniformisation $F \subseteq R$

Projection $\pi(F) = \pi(R)$

Theorem [Axiom of Choice]

Every relation admits a uniformisation.

Uniformisation



Relation $R \subseteq X \times Y$

Uniformisation $F \subseteq R$

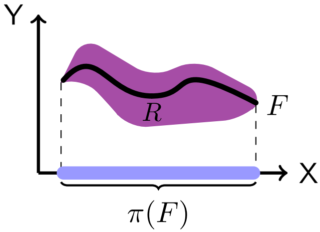
Projection $\pi(F) = \pi(R)$

Theorem [Axiom of Choice]

Every relation admits a uniformisation.

What about definability?

Uniformisation



Relation $R \subseteq X \times Y$

Uniformisation $F \subseteq R$

Projection $\pi(F) = \pi(R)$

Theorem [Axiom of Choice]

Every relation admits a uniformisation.

What about definability?

Theorem (Novikov, Kondô [1938])

Every co-analytic ($\mathbf{\Pi}^1_1$) relation admits a co-analytic uniformisation.

Structures

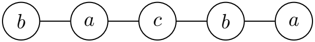
Structures over an alphabet A — a finite set of symbols

Structures over an alphabet A — a finite set of symbols

Words:

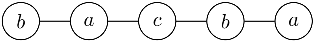
Structures over an alphabet A — a finite set of symbols

Words:

$w =$ 

Structures over an alphabet A — a finite set of symbols

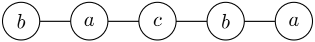
Words:

$w =$ 

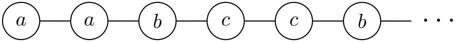
$w\colon \{1, \ldots, |w|\} \to A$

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ 

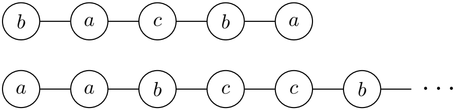
$w \colon \{1, \ldots, |w|\} \to A$

$\alpha =$ 

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ $\alpha =$

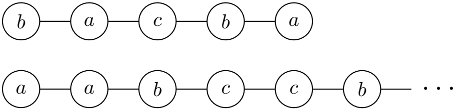


$w\colon \{1, \ldots, |w|\} \to A$

$\alpha\colon \omega \to A$

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ 

$w\colon \{1, \ldots, |w|\} \to A$

$\alpha =$

$\alpha\colon \omega \to A$

Signature: $\leqslant$, $s(x)$, $a(x)$ for $a \in A$

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ (b) — (a) — (c) — (b) — (a) $\qquad w\colon \{1, \ldots, |w|\} \to A$

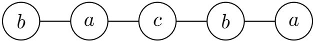
$\alpha =$ (a) — (a) — (b) — (c) — (c) — (b) — $\cdots \qquad \alpha\colon \omega \to A$

Signature: $\leqslant$, $s(x)$, $a(x)$ for $a \in A$

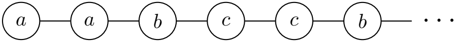
Trees:

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ 

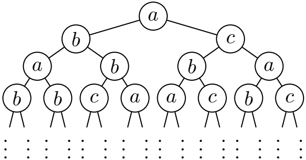
$w\colon \{1, \ldots, |w|\} \to A$

$\alpha =$ 

$\alpha\colon \omega \to A$

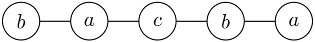
Signature: $\leqslant$, $s(x)$, $a(x)$ for $a \in A$

Trees:

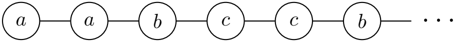
$t =$ 

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ 

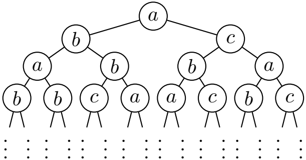
$w\colon \{1, \ldots, |w|\} \to A$

$\alpha =$ 

$\alpha\colon \omega \to A$

Signature: $\leqslant$, $s(x)$, $a(x)$ for $a \in A$

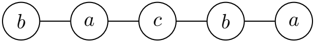
Trees:

$t =$ 

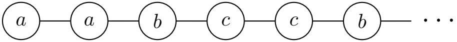
$t\colon \{0,1\}^{<\omega} \to A$

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ 

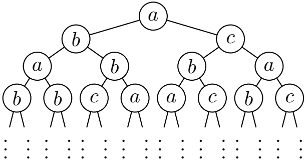
$w\colon \{1, \ldots, |w|\} \to A$

$\alpha =$ 

$\alpha\colon \omega \to A$

Signature: $\leqslant$, $s(x)$, $a(x)$ for $a \in A$

Trees:

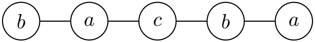
$t =$ 

$t\colon \{0,1\}^{<\omega} \to A$

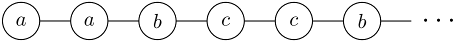
Signature: $\preceq$, $s_0(x)$, $s_1(x)$, $a(x)$ for $a \in A$

Structures over an alphabet A — a finite set of symbols

Words:

$w =$ 

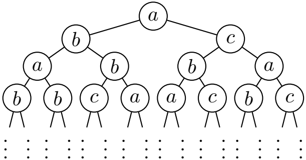
$w\colon \{1, \ldots, |w|\} \to A$

$\alpha =$ 

$\alpha\colon \omega \to A$

Signature: $\leqslant$, $s(x)$, $a(x)$ for $a \in A$

Trees:

$t =$ 

$t\colon \{0,1\}^{<\omega} \to A$

Signature: $\preceq$, $s_0(x)$, $s_1(x)$, $a(x)$ for $a \in A$

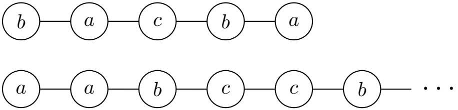
Pairing: $s = abc$ over A, $s' = xyz$ over A' [and $\mathrm{dom}(s) = \mathrm{dom}(s')$]

Structures over an alphabet A — a finite set of symbols

Words:

$w =$

$\alpha =$



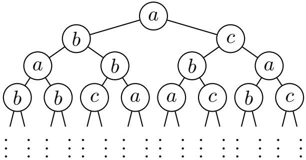
$w\colon \{1, \ldots, |w|\} \to A$

$\alpha\colon \omega \to A$

Signature: $\leqslant$, $s(x)$, $a(x)$ for $a \in A$

Trees:

$t =$



$t\colon \{0,1\}^{<\omega} \to A$

Signature: $\preceq$, $s_0(x)$, $s_1(x)$, $a(x)$ for $a \in A$

Pairing: $s = abc$ over A, $s' = xyz$ over A' $\big[$and $\mathrm{dom}(s) = \mathrm{dom}(s')\big]$

$(s, s') \sim s \otimes s' = \begin{smallmatrix} a & b & c \\ x & y & z \end{smallmatrix}$ over $A \times A'$

Logic

Logic

First-order (FO) logic:

Logic

First-order (FO) logic:

$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi$ predicates

Logic

First-order (FO) logic:

$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi$ predicates

$\big[x, y$ — nodes of the structure$\big]$

Logic

First-order (FO) logic:

$$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi \qquad \text{predicates}$$

$\big[x,\, y$ — nodes of the structure$\big]$

Monadic second-order (MSO) logic:

Logic

First-order (FO) logic:

$$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi \qquad \text{predicates}$$

$\big[x, y$ — nodes of the structure$\big]$

Monadic second-order (MSO) logic:

$$+\ \exists_X \quad \forall_X \quad x \in X$$

Logic

First-order (FO) logic:

$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi$ predicates

[x, y — nodes of the structure]

Monadic second-order (MSO) logic:

$+ \; \exists_X \quad \forall_X \quad x \in X$

[X, Y — sets of nodes of the structure]

Logic

First-order (FO) logic:

$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi$ predicates

$\big[x$, y — nodes of the structure$\big]$

Monadic second-order (MSO) logic:

$+ \; \exists_X \quad \forall_X \quad x \in X$

$\big[X$, Y — sets of nodes of the structure$\big]$

$\rightsquigarrow$ expressive power subsuming LTL, CTL*, modal μ-calculus, ...

Logic

First-order (FO) logic:

$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi$ predicates

[x, y — nodes of the structure]

Monadic second-order (MSO) logic:

$+ \; \exists_X \quad \forall_X \quad x \in X$

[X, Y — sets of nodes of the structure]

$\rightsquigarrow$ expressive power subsuming LTL, CTL*, modal μ-calculus, ...

Theorem (Rabin [1969])

The satisfiability problem is decidable for MSO over infinite trees.

Logic

First-order (FO) logic:

$\exists_x \quad \forall_x \quad \neg\psi \quad \varphi \vee \psi \quad \varphi \wedge \psi$ predicates

[x, y — nodes of the structure]

Monadic second-order (MSO) logic:

$+ \; \exists_X \quad \forall_X \quad x \in X$

[X, Y — sets of nodes of the structure]

$\rightsquigarrow$ expressive power subsuming LTL, CTL*, modal μ-calculus, ...

Theorem (Rabin [1969])

The satisfiability problem is decidable for MSO over infinite trees.

$\rightsquigarrow$ applications to verification and model-checking

Uniformisability in FO/ MSO logic?

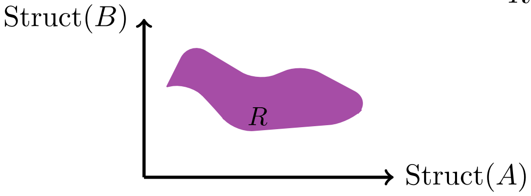
Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$

Uniformisability in FO/ MSO logic?

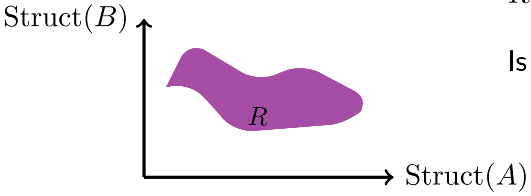
Fix a formula φ over $A \times B$

$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$



Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$

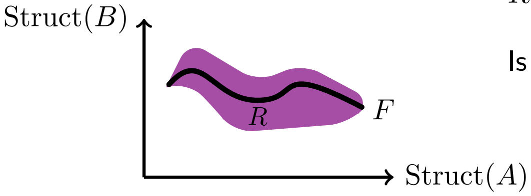


$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

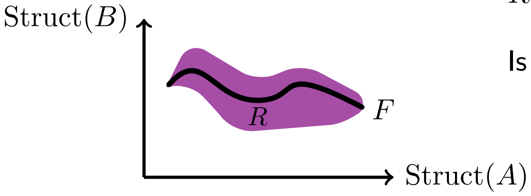
Is there ψ such that

$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

uniformises R?

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

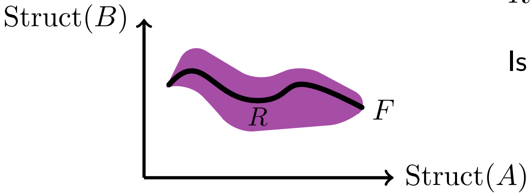
$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

uniformises R?

Technical assumption: we restrict φ to (s, s') s.t. $\mathrm{dom}(s) = \mathrm{dom}(s')$

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

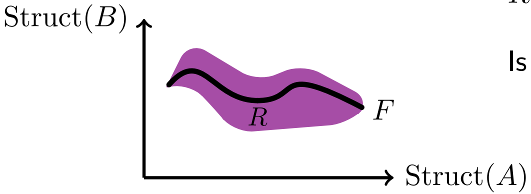
uniformises R?

Technical assumption: we restrict φ to (s, s') s.t. $\mathrm{dom}(s) = \mathrm{dom}(s')$

Variants:

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

uniformises R?

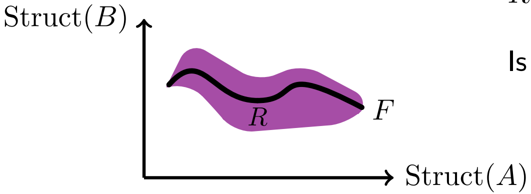
Technical assumption: we restrict φ to (s, s') s.t. $\mathrm{dom}(s) = \mathrm{dom}(s')$

Variants:

— φ, ψ in FO / MSO?

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

uniformises R?

Technical assumption: we restrict φ to (s, s') s.t. $\mathrm{dom}(s) = \mathrm{dom}(s')$

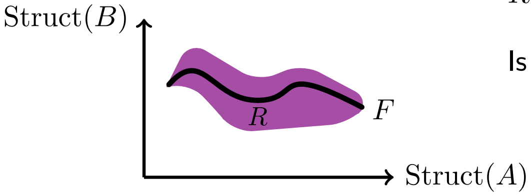
Variants:

— φ, ψ in FO / MSO?

— Struct are: **finite**/**infinite** **words**/**trees**?

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

uniformises R?

Technical assumption: we restrict φ to (s, s') s.t. $\mathrm{dom}(s) = \mathrm{dom}(s')$

Variants:

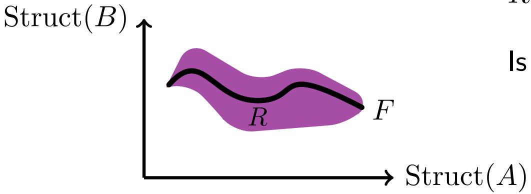
— φ, ψ in FO / MSO?

— Struct are: **finite**/**infinite** **words**/**trees**?

— ψ may use additional monadic parameters $P_1, \ldots, P_n$:

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

uniformises R?

Technical assumption: we restrict φ to (s, s') s.t. $\mathrm{dom}(s) = \mathrm{dom}(s')$

Variants:

— φ, ψ in FO / MSO?

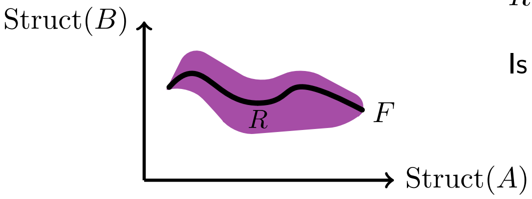
— Struct are: **finite**/**infinite** **words**/**trees**?

— ψ may use additional monadic parameters $P_1, \ldots, P_n$:

$$F = \{(s, s') \mid s \otimes s' \models \psi(P_1, \ldots, P_n)\}?$$

Uniformisability in FO/ MSO logic?

Fix a formula φ over $A \times B$



$$R = \{(s, s') \mid s \otimes s' \models \varphi\}$$

Is there ψ such that

$$F = \{(s, s') \mid s \otimes s' \models \psi\}$$

uniformises R?

Technical assumption: we restrict φ to (s, s') s.t. $\mathrm{dom}(s) = \mathrm{dom}(s')$

Variants:

— φ, ψ in FO / MSO?

— Struct are: **finite**/**infinite** **words**/**trees**?

— ψ may use additional monadic parameters $P_1, \ldots, P_n$:

$$F = \{(s, s') \mid s \otimes s' \models \psi(P_1, \ldots, P_n)\}?$$

— ψ can be effectively constructed from φ?

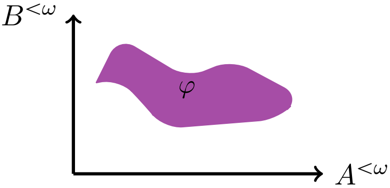
MSO over finite words ✓

MSO over finite words ✓

Take φ over $A \times B$

MSO over finite words ✓

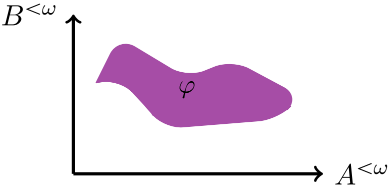
Take φ over $A \times B$



MSO over finite words ✓

Take φ over $A \times B$

Let $(s, s') \in F$ if:

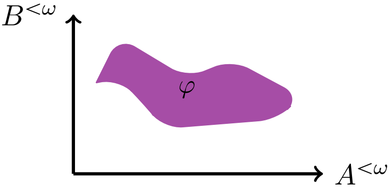


MSO over finite words ✓

Take φ over $A \times B$

Let $(s, s') \in F$ if:

— $s \otimes s' \models \varphi$



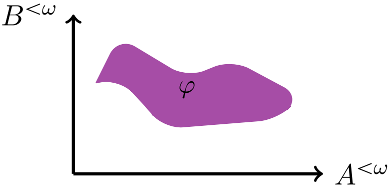
MSO over finite words ✓

Take φ over $A \times B$

Let $(s, s') \in F$ if:

— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$



MSO **over finite words ✓**

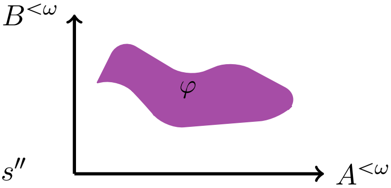
Take φ over $A \times B$

Let $(s, s') \in F$ if:

— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''



MSO over finite words ✓

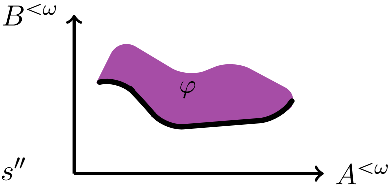
Take φ over $A \times B$

Let $(s, s') \in F$ if:

— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''



MSO over finite words ✓

Take φ over $A \times B$

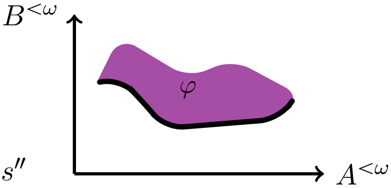
Let $(s, s') \in F$ if:

— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''

$\rightsquigarrow$ F is effectively MSO-definable



MSO **over finite words ✓**

Take φ over $A \times B$

Let $(s, s') \in F$ if:

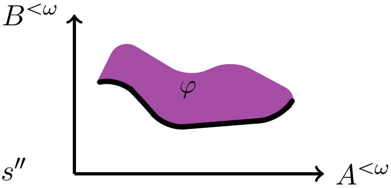
— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''

$\rightsquigarrow$ F is effectively MSO-definable

MSO **over finite trees ✓**



MSO over finite words ✓

Take φ over $A \times B$

Let $(s, s') \in F$ if:

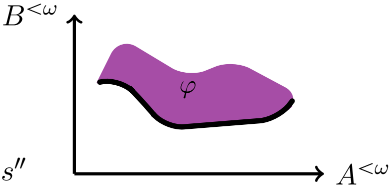
— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''

$\rightsquigarrow$ F is effectively MSO-definable

MSO over finite trees ✓



MSO over infinite words ?

MSO over finite words ✓

Take φ over $A \times B$

Let $(s, s') \in F$ if:

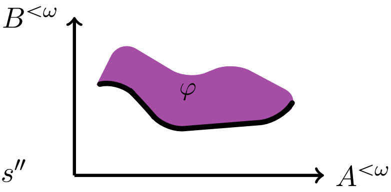
— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''

$\rightsquigarrow$ F is effectively MSO-definable

MSO over finite trees ✓



MSO over infinite words ?

Problem: there may be no lexicographically minimal witness:

MSO **over finite words ✓**

Take φ over $A \times B$

Let $(s, s') \in F$ if:

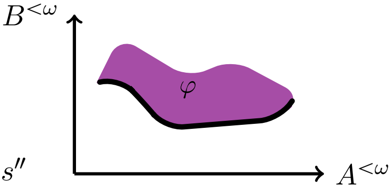
— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''

$\rightsquigarrow$ F is effectively MSO-definable

MSO **over finite trees ✓**



MSO **over infinite words ?**

Problem: there may be no lexicographically minimal witness:

$$\varphi \equiv \text{“}s' \text{ has finitely many } a\text{”}$$

MSO over finite words ✓

Take φ over $A \times B$

Let $(s, s') \in F$ if:

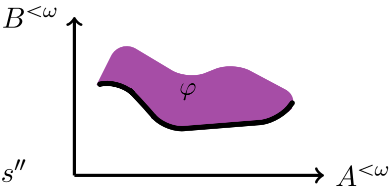
— $s \otimes s' \models \varphi$

— for every s'' such that $s \otimes s'' \models \varphi$

s' is lexicographically smaller than s''

$\rightsquigarrow$ F is effectively MSO-definable

MSO over finite trees ✓



MSO over infinite words ?

Problem: there may be no lexicographically minimal witness:

$$\varphi \equiv \text{“}s' \text{ has finitely many } a\text{”}$$

$$(bbbb\ldots) \geqslant_{\text{lex}} (abbb\ldots) \geqslant_{\text{lex}} (aabb\ldots) \geqslant_{\text{lex}} (aaab\ldots) \geqslant_{\text{lex}} \ldots$$

MSO over infinite words ?

MSO **over infinite words ?**

Lifsches, Shelah, JSL [1998]

THEOREM 6.3. $(\omega, <)$ *has the uniformization property.*

PROOF. By [1]. ⊣

REFERENCES

[1] J. R. BÜCHI and L. H. LANDWEBER, *Solving sequential conditions by finite-state strategies*, ***Transactions of the American Mathematical Society***, vol. 138 (1969), pp. 295–311.

MSO **over infinite words ?**

Lifsches, Shelah, JSL [1998]

THEOREM 6.3. $(\omega, <)$ *has the uniformization property.*

PROOF. By [1]. ⊣

REFERENCES

[1] J. R. BÜCHI and L. H. LANDWEBER, *Solving sequential conditions by finite-state strategies*, ***Transactions of the American Mathematical Society***, vol. 138 (1969), pp. 295–311.

Theorem (Siefkes [1975], Rabinovich [2007])

MSO has effective uniformisation property over infinite words.

MSO over infinite words ?

Lifsches, Shelah, JSL [1998]

THEOREM 6.3. $(\omega, <)$ *has the uniformization property.*

PROOF. By [1]. ⊣

REFERENCES

[1] J. R. BÜCHI and L. H. LANDWEBER, *Solving sequential conditions by finite-state strategies*, ***Transactions of the American Mathematical Society***, vol. 138 (1969), pp. 295–311.

Theorem (Siefkes [1975], Rabinovich [2007])

MSO has effective uniformisation property over infinite words.

Proof

Uniformise accepting runs of a non-deterministc Büchi automaton equivalent to φ:

MSO over infinite words ?

Lifsches, Shelah, JSL [1998]

THEOREM 6.3. $(\omega, <)$ *has the uniformization property.*

PROOF. By [1]. ⊣

REFERENCES

[1] J. R. BÜCHI and L. H. LANDWEBER, *Solving sequential conditions by finite-state strategies*, ***Transactions of the American Mathematical Society***, vol. 138 (1969), pp. 295–311.

Theorem (Siefkes [1975], Rabinovich [2007])

MSO has effective uniformisation property over infinite words.

Proof

Uniformise accepting runs of a non-deterministc Büchi automaton equivalent to φ:

pick the $\leqslant_{\text{lex}}$-minimal accepting run $\big(F <_{\text{lex}} (Q{-}F)\big)$ ■

MSO over infinite words ✓

Lifsches, Shelah, JSL [1998]

THEOREM 6.3. $(\omega, <)$ *has the uniformization property.*

PROOF. By [1]. ⊣

REFERENCES

[1] J. R. BÜCHI and L. H. LANDWEBER, *Solving sequential conditions by finite-state strategies*, ***Transactions of the American Mathematical Society***, vol. 138 (1969), pp. 295–311.

Theorem (Siefkes [1975], Rabinovich [2007])

MSO has effective uniformisation property over infinite words.

Proof

Uniformise accepting runs of a non-deterministc Büchi automaton equivalent to φ:

pick the $\leqslant_{\text{lex}}$-minimal accepting run $\big(F <_{\text{lex}} (Q{-}F)\big)$ ■

What about FO?

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$a\ a\ a\ a \cdots a\ a\ \sharp\ a\ a\ a\ a \cdots a\ a$

$0\ 1\ 0\ 1 \cdots 0\ 1\ \sharp\ ?\ ?\ ?\ ? \cdots ?\ ?$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$a\ a\ a\ a \cdots a\ a\ \sharp\ a\ a\ a\ a \cdots a\ a$$

$$0\ 1\ 0\ 1 \cdots 0\ 1\ \sharp\ ?\ ?\ ?\ ? \cdots ?\ ?$$

i.e. $\binom{a\ a}{0\ 1}^* \ \binom{\sharp}{\sharp}\ \binom{a}{?}^*$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix}$$ **and**

i.e. $\binom{a\ a}{0\ 1}^* \ \binom{\sharp}{\sharp} \ \binom{a}{?}^*$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

i.e. $\binom{a\,a}{0\,1}^* \, {\sharp \atop \sharp} \, \binom{a}{?}^*$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

i.e. $\binom{a\,a}{0\,1}^* \begin{smallmatrix}\sharp\\ \sharp\end{smallmatrix} \binom{a}{?}^*$ i.e. $\binom{a}{?}^* \begin{smallmatrix}\sharp\\ \sharp\end{smallmatrix} \binom{a\,a}{0\,1}^*$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

$\Big[$both $\begin{smallmatrix} a & a & \sharp & a & a \\ ? & ? & \sharp & 0 & 1 \end{smallmatrix}$ and $\begin{smallmatrix} a & a & \sharp & a & a \\ 0 & 1 & \sharp & ? & ? \end{smallmatrix}$ belong to $R\Big]$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

Assume that an FO-definable F uniformises R.

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

Assume that an FO-definable F uniformises R.

Take n big enough s.t. $uw^n v \in F$ **iff** $uw^{n+1} v \in F$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

Assume that an FO-definable F uniformises R.

Take n big enough s.t. $uw^n v \in F$ **iff** $uw^{n+1} v \in F$

$$\binom{a}{?}^{2n+1} \binom{\sharp}{\sharp} \binom{a\ a}{0\ 1}^{n} \in F \quad \text{and} \quad \binom{a\ a}{0\ 1}^{n} \binom{\sharp}{\sharp} \binom{a}{?}^{2n+1} \in F$$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

Assume that an FO-definable F uniformises R.

Take n big enough s.t. $uw^nv \in F$ **iff** $uw^{n+1}v \in F$

$$\binom{a}{?}^{2n+1} \binom{\sharp}{\sharp} \binom{a\,a}{0\,1}^{n} \in F \quad \text{and} \quad \binom{a\,a}{0\,1}^{n} \binom{\sharp}{\sharp} \binom{a}{?}^{2n+1} \in F$$

$$\binom{a}{?}^{2n} \binom{\sharp}{\sharp} \binom{a\,a}{0\,1}^{n} \in F \quad \text{and} \quad \binom{a\,a}{0\,1}^{n} \binom{\sharp}{\sharp} \binom{a}{?}^{2n} \in F$$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

Assume that an FO-definable F uniformises R.

Take n big enough s.t. $uw^n v \in F$ **iff** $uw^{n+1} v \in F$

$$\binom{a}{?}^{2n+1} \binom{\sharp}{\sharp} \binom{a\ a}{0\ 1}^{n} \in F \quad \text{and} \quad \binom{a\ a}{0\ 1}^{n} \binom{\sharp}{\sharp} \binom{a}{?}^{2n+1} \in F$$

$$\binom{a}{?}^{2n} \binom{\sharp}{\sharp} \binom{a\ a}{0\ 1}^{n} \in F \quad \text{and} \quad \binom{a\ a}{0\ 1}^{n} \binom{\sharp}{\sharp} \binom{a}{?}^{2n} \in F$$

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

Assume that an FO-definable F uniformises R.

Take n big enough s.t. $uw^n v \in F$ **iff** $uw^{n+1} v \in F$

$$\binom{a}{?}^{2n+1} \binom{\sharp}{\sharp} \binom{a\ a}{0\ 1}^{n} \in F \quad \text{and} \quad \binom{a\ a}{0\ 1}^{n} \binom{\sharp}{\sharp} \binom{a}{?}^{2n+1} \in F$$

$$\binom{a}{?}^{2n} \binom{\sharp}{\sharp} \binom{a\ a}{0\ 1}^{n} \in F \quad \text{and} \quad \binom{a\ a}{0\ 1}^{n} \binom{\sharp}{\sharp} \binom{a}{?}^{2n} \in F$$

$\rightsquigarrow$ F is not uniformised! ■

What about FO?

Theorem (?)

FO over finite words does **not** have uniformisation property.

Proof

Consider $A = \{a, \sharp\}$, $B = \{0, 1, ?, \sharp\}$, and R containing:

$$\begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \sharp & ? & ? & ? & ? & \cdots & ? & ? \end{matrix} \quad \textbf{and} \quad \begin{matrix} a & a & a & a & \cdots & a & a & \sharp & a & a & a & a & \cdots & a & a \\ ? & ? & ? & ? & \cdots & ? & ? & \sharp & 0 & 1 & 0 & 1 & \cdots & 0 & 1 \end{matrix}$$

Assume that an FO-definable F uniformises R.

Take n big enough s.t. $uw^n v \in F$ **iff** $uw^{n+1} v \in F$

$$\binom{a}{?}^{2n+1} \begin{matrix}\sharp\\ \sharp\end{matrix} \binom{a\ a}{0\ 1}^{n} \in F \quad \text{and} \quad \binom{a\ a}{0\ 1}^{n} \begin{matrix}\sharp\\ \sharp\end{matrix} \binom{a}{?}^{2n+1} \in F$$

$$\binom{a}{?}^{2n} \begin{matrix}\sharp\\ \sharp\end{matrix} \binom{a\ a}{0\ 1}^{n} \in F \quad \text{and} \quad \binom{a\ a}{0\ 1}^{n} \begin{matrix}\sharp\\ \sharp\end{matrix} \binom{a}{?}^{2n} \in F$$

⇝ F is not uniformised! ∎

⇝ **no** uniformisation in FO over **finite/infinite words/trees**

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Theorem (Gurevich, Shelah [1983])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Theorem (Gurevich, Shelah [1983])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

I.e. there is no $\psi(\mathbf{X}, \mathbf{y})$ such that:

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Theorem (Gurevich, Shelah [1983])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

I.e. there is no $\psi(\mathbf{X}, \mathbf{y})$ such that:

$$\forall_{\emptyset \neq \mathbf{X} \subseteq \{0,1\}^{<\omega}} \quad \exists!_{\mathbf{y} \in \mathbf{X}} \quad \psi(\mathbf{X}, \mathbf{y})$$

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Theorem (Gurevich, Shelah [1983])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

I.e. there is no $\psi(\mathbf{X}, \mathbf{y})$ such that:

$$\forall_{\emptyset \neq \mathbf{X} \subseteq \{0,1\}^{<\omega}} \quad \exists!_{\mathbf{y} \in \mathbf{X}} \quad \psi(\mathbf{X}, \mathbf{y})$$

$\rightsquigarrow$ "no MSO-def. choice function over infinite trees"

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Theorem (Gurevich, Shelah [1983])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

I.e. there is no $\psi(\mathbf{X}, \mathbf{y})$ such that:

$$\forall_{\emptyset \neq \mathbf{X} \subseteq \{0,1\}^{<\omega}} \quad \exists!_{\mathbf{y} \in \mathbf{X}} \quad \psi(\mathbf{X}, \mathbf{y})$$

$\rightsquigarrow$ "no MSO-def. choice function over infinite trees"

Proof

A forcing-based argument. [with some subtleties]

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Theorem (Gurevich, Shelah [1983])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

I.e. there is no $\psi(\mathbf{X}, \mathbf{y})$ such that:

$$\forall_{\emptyset \neq \mathbf{X} \subseteq \{0,1\}^{<\omega}} \quad \exists !_{\mathbf{y} \in \mathbf{X}} \quad \psi(\mathbf{X}, \mathbf{y})$$

⇝ "no MSO-def. choice function over infinite trees"

Proof

A forcing-based argument. [with some subtleties]

Theorem (Carayol, Löding [2007])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

Rabin's Uniformisation Problem

Does MSO have uniformisation property over infinite trees?

Theorem (Gurevich, Shelah [1983])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

I.e. there is no $\psi(\mathbf{X}, \mathbf{y})$ such that:

$$\forall_{\emptyset \neq \mathbf{X} \subseteq \{0,1\}^{<\omega}} \quad \exists!_{\mathbf{y} \in \mathbf{X}} \quad \psi(\mathbf{X}, \mathbf{y})$$

$\rightsquigarrow$ "no MSO-def. choice function over infinite trees"

Proof

A forcing-based argument. [with some subtleties]

Theorem (Carayol, Löding [2007])

The relation $\mathbf{y} \in \mathbf{X}$ does not admit MSO-def. uniformisation of $\mathbf{y}$.

Proof

Pumping of runs of a marking automaton. ■

Parameters

Parameters

Theorem

Parameters do not help over the complete binary tree.

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

$$\Big\{(s,s') \mid s \otimes s' \models \psi\big(\vec{P}\big)\Big\} \text{ uniformises } \Big\{(s,s') \mid s \otimes s' \models \varphi\Big\}$$

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

$$\Big\{(s,s') \mid s\otimes s' \models \psi\big(\vec{P}\big)\Big\} \text{ uniformises } \Big\{(s,s') \mid s\otimes s' \models \varphi\Big\}$$

$\big[$shortly: $\psi(\vec{P})$ uniformises $\varphi\big]$

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

$$\Big\{(s,s') \mid s \otimes s' \models \psi\big(\vec{P}\big)\Big\} \text{ uniformises } \Big\{(s,s') \mid s \otimes s' \models \varphi\Big\}$$

$$\big[\text{shortly: } \psi(\vec{P}) \text{ uniformises } \varphi\big]$$

Consider $L = \Big\{\vec{P} \mid \psi\big(\vec{P}\big) \text{ uniformises } \varphi\Big\}$

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

$$\Big\{(s,s') \mid s \otimes s' \models \psi\big(\vec{P}\big)\Big\} \text{ uniformises } \Big\{(s,s') \mid s \otimes s' \models \varphi\Big\}$$

$$\big[\text{shortly: } \psi(\vec{P}) \text{ uniformises } \varphi\big]$$

Consider $L = \Big\{\vec{P} \mid \psi\big(\vec{P}\big) \text{ uniformises } \varphi\Big\}$

L is MSO-definable

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

$$\Big\{(s,s') \mid s\otimes s' \models \psi\big(\vec{P}\big)\Big\} \text{ uniformises } \Big\{(s,s') \mid s\otimes s' \models \varphi\Big\}$$

$$\big[\text{shortly: } \psi\big(\vec{P}\big) \text{ uniformises } \varphi\big]$$

Consider $L = \Big\{\vec{P} \mid \psi\big(\vec{P}\big) \text{ uniformises } \varphi\Big\}$

L is MSO-definable $\rightsquigarrow$ there is an MSO-definable $\overrightarrow{P_0} \in L$

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

$$\Big\{(s,s') \mid s \otimes s' \models \psi\big(\vec{P}\big)\Big\} \text{ uniformises } \Big\{(s,s') \mid s \otimes s' \models \varphi\Big\}$$

[shortly: $\psi\big(\vec{P}\big)$ uniformises φ]

Consider $L = \Big\{\vec{P} \mid \psi\big(\vec{P}\big) \text{ uniformises } \varphi\Big\}$

L is MSO-definable $\rightsquigarrow$ there is an MSO-definable $\overrightarrow{P_0} \in L$

Consider $\psi_0 = \psi\big(\overrightarrow{P_0}\big)$

Parameters

Theorem

Parameters do not help over the complete binary tree.

Proof

Assume that $\vec{P}$ is a tuple of subsets of $\{0,1\}^{<\omega}$ such that:

$$\left\{(s,s') \mid s \otimes s' \models \psi(\vec{P})\right\} \text{ uniformises } \left\{(s,s') \mid s \otimes s' \models \varphi\right\}$$

$$\left[\text{shortly: } \psi(\vec{P}) \text{ uniformises } \varphi\right]$$

Consider $L = \left\{\vec{P} \mid \psi(\vec{P}) \text{ uniformises } \varphi\right\}$

L is MSO-definable $\rightsquigarrow$ there is an MSO-definable $\overrightarrow{P_0} \in L$

Consider $\psi_0 = \psi(\overrightarrow{P_0})$

ψ_0 has **no** parameters and uniformises φ ■

MSO **over non-complete trees** (with parameters)

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

$$\left[\tau \sim \text{a closed subset of } \{0,1\}^{\omega}\right]$$

MSO **over non-complete trees** (with parameters)

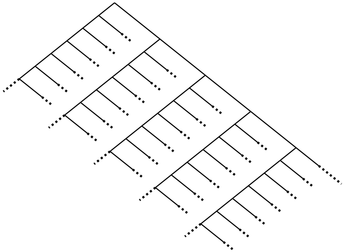
Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

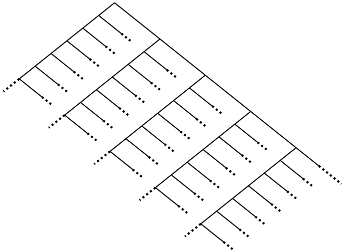


MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"



MSO **over non-complete trees** (with parameters)

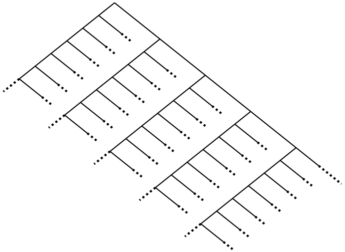
Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"

Theorem (Lifsches, Shelah [1998], S. [2013])

[adjusted to $\{0,1\}^{<\omega}$]

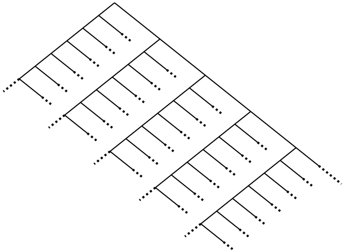


MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"



Theorem (Lifsches, Shelah [1998], S. [2013])

[adjusted to $\{0,1\}^{<\omega}$]

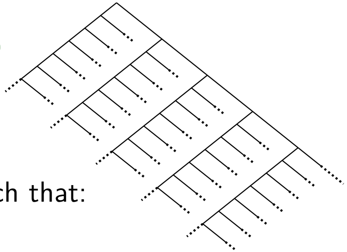
For every prefix-closed $\tau \subseteq \{0,1\}^{<\omega}$ either:

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"



Theorem (Lifsches, Shelah [1998], S. [2013])

[adjusted to $\{0,1\}^{<\omega}$]

For every prefix-closed $\tau \subseteq \{0,1\}^{<\omega}$ either:

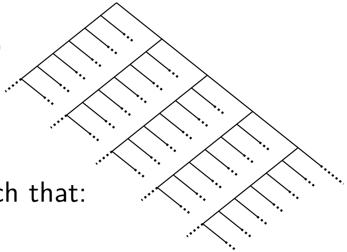
- τ is scattered and there exists $P \subseteq \tau$ such that:

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"



Theorem (Lifsches, Shelah [1998], S. [2013])

[adjusted to $\{0,1\}^{<\omega}$]

For every prefix-closed $\tau \subseteq \{0,1\}^{<\omega}$ either:

- τ is scattered and there exists $P \subseteq \tau$ such that:

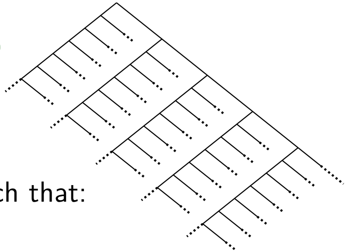
  $\forall_\varphi \;\; \exists_\psi \;\; \psi(P)$ uniformises φ over τ

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"



Theorem (Lifsches, Shelah [1998], S. [2013])

[adjusted to $\{0,1\}^{<\omega}$]

For every prefix-closed $\tau \subseteq \{0,1\}^{<\omega}$ either:

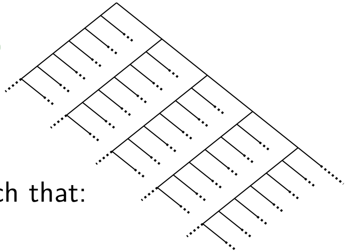
- τ is scattered and there exists $P \subseteq \tau$ such that:
  $\forall_{\varphi} \;\; \exists_{\psi} \;\; \psi(P)$ uniformises φ over τ
- τ contains a complete binary subtree (a perfect set) and:

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"



Theorem (Lifsches, Shelah [1998], S. [2013])

[adjusted to $\{0,1\}^{<\omega}$]

For every prefix-closed $\tau \subseteq \{0,1\}^{<\omega}$ either:

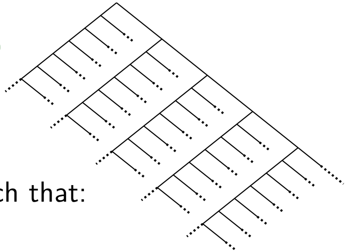
- τ is scattered and there exists $P \subseteq \tau$ such that:
  $\forall_{\varphi}\ \exists_{\psi}\ \psi(P)$ uniformises φ over τ
- τ contains a complete binary subtree (a perfect set) and:
  there exists φ non-uniformisable over τ (even with parameters)

MSO **over non-complete trees** (with parameters)

Prefix-closed sets $\tau \subseteq \{0,1\}^{<\omega}$ and labellings $t\colon \tau \to A$

τ is scattered if it has only countably many branches

"Scattered trees = Finite trees $\otimes$ Infinite words"



Theorem (Lifsches, Shelah [1998], S. [2013])

[adjusted to $\{0,1\}^{<\omega}$]

For every prefix-closed $\tau \subseteq \{0,1\}^{<\omega}$ either:

- τ is scattered and there exists $P \subseteq \tau$ such that:
  $\forall_{\varphi} \;\; \exists_{\psi} \;\; \psi(P)$ uniformises φ over τ
- τ contains a complete binary subtree (a perfect set) and:
  there exists φ non-uniformisable over τ (even with parameters)

$\rightsquigarrow$ a complete characterisation (with parameters depending on τ)

MSO, scattered trees, **no** parameters **?**

MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ
into separate branches

MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

into separate branches

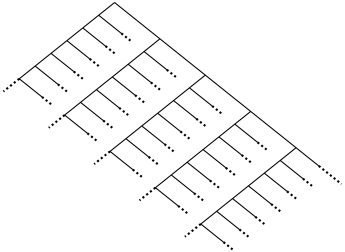
[formally a subset of τ]

MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

into separate branches

[formally a subset of τ]

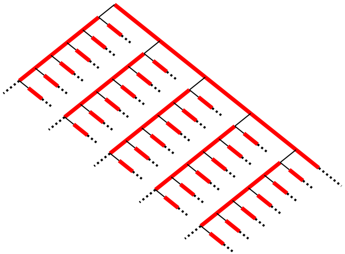


MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

into separate branches

[formally a subset of τ]

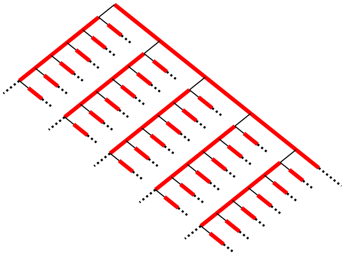


MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

into separate branches

[formally a subset of τ]



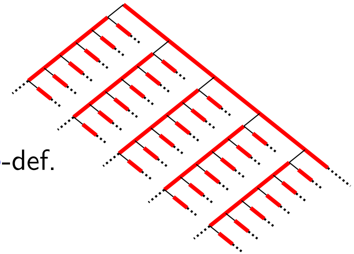
1. τ has a skeleton $\iff$ τ is scattered

MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

into separate branches

[formally a subset of τ]



1. τ has a skeleton $\iff$ τ is scattered

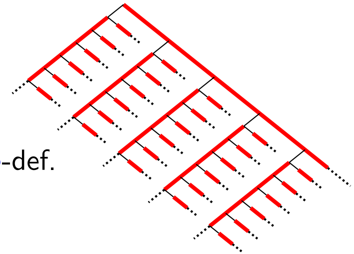
2. $R = \{(\tau, \sigma) \mid \sigma$ is a skeleton of $\tau\}$ is MSO-def.

MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

into separate branches

[formally a subset of τ]



1. τ has a skeleton $\Longleftrightarrow$ τ is scattered

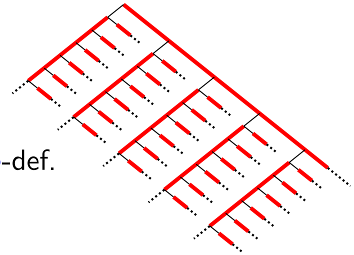
2. $R = \{(\tau, \sigma) \mid \sigma \text{ is a skeleton of } \tau\}$ is MSO-def.

MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

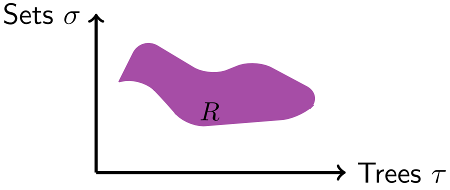
into separate branches

[formally a subset of τ]



1. τ has a skeleton $\iff$ τ is scattered

2. $R = \big\{(\tau, \sigma) \mid \sigma \text{ is a skeleton of } \tau\big\}$ is MSO-def.

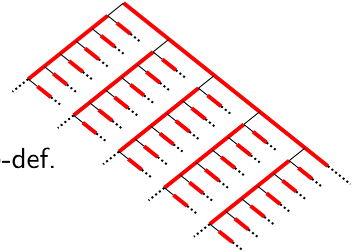


MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

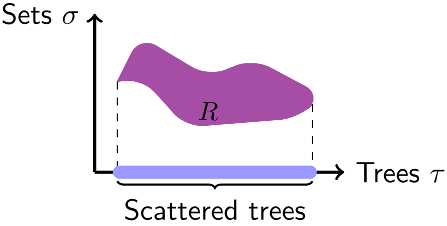
into separate branches

[formally a subset of τ]



1. τ has a skeleton $\iff$ τ is scattered

2. $R = \{(\tau, \sigma) \mid \sigma$ is a skeleton of $\tau\}$ is MSO-def.

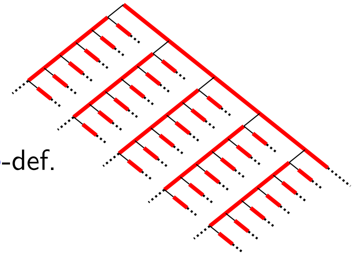


MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

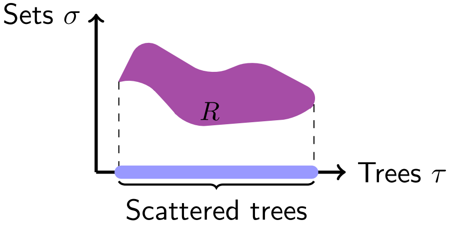
into separate branches

[formally a subset of τ]



1. τ has a skeleton $\iff$ τ is scattered

2. $R = \{(\tau, \sigma) \mid \sigma$ is a skeleton of $\tau\}$ is MSO-def.



Theorem (S. [2013])

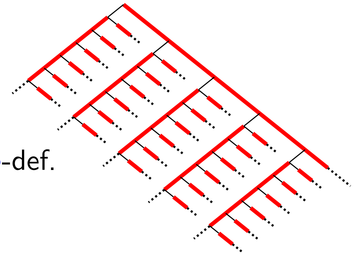
There is **no** MSO-def. uniformisation of R.

MSO, scattered trees, **no** parameters **?**

Skeleton = well-founded decomposition of a scattered τ

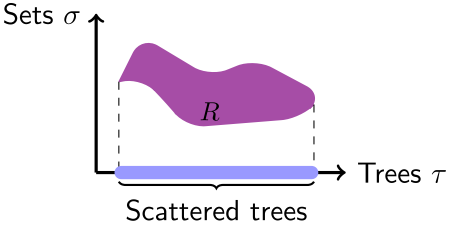
into separate branches

[formally a subset of τ]



1. τ has a skeleton $\iff$ τ is scattered

2. $R = \{(\tau, \sigma) \mid \sigma \text{ is a skeleton of } \tau\}$ is MSO-def.



Theorem (S. [2013])

There is **no** MSO-def. uniformisation of R.

$\leadsto$ new non-uniformisability example

Choice on scattered trees

Choice on scattered trees

Conjecture (S. [2013])

The relation "$\mathbf{y} \in \mathbf{X}$ and $\mathbf{X}$ is contained in a scattered tree" does not admit MSO-def. uniformisation of $\mathbf{y}$ (without parameters).

Choice on scattered trees

Conjecture (S. [2013])

The relation "$\mathbf{y} \in \mathbf{X}$ and $\mathbf{X}$ is contained in a scattered tree" does not admit MSO-def. uniformisation of $\mathbf{y}$ (without parameters).

$\leftrightsquigarrow$ no MSO-def. choice function on scattered trees

Choice on scattered trees

Conjecture (S. [2013])

The relation "$\mathbf{y} \in \mathbf{X}$ and $\mathbf{X}$ is contained in a scattered tree" does not admit MSO-def. uniformisation of $\mathbf{y}$ (without parameters).

$\leftrightsquigarrow$ no MSO-def. choice function on scattered trees

A side effect of a study on *thin algebras*. . .

Choice on scattered trees

Conjecture (S. [2013])

The relation "$\mathbf{y} \in \mathbf{X}$ and $\mathbf{X}$ is contained in a scattered tree" does not admit MSO-def. uniformisation of $\mathbf{y}$ (without parameters).

$\leftrightsquigarrow$ no MSO-def. choice function on scattered trees

A side effect of a study on *thin algebras*. . .

Theorem (Bojańczyk, Idziaszek, S. [2013])

MSO over scattered (aka *thin*) trees is equivalent with thin algebra.

Choice on scattered trees

Conjecture (S. [2013])

The relation "$\mathbf{y} \in \mathbf{X}$ and $\mathbf{X}$ is contained in a scattered tree" does not admit MSO-def. uniformisation of $\mathbf{y}$ (without parameters).

$\leftrightsquigarrow$ no MSO-def. choice function on scattered trees

A side effect of a study on *thin algebras*...

Theorem (Bojańczyk, Idziaszek, S. [2013])

MSO over scattered (aka *thin*) trees is equivalent with thin algebra.

$\rightsquigarrow$ effective characterisations (weak MSO, ...)

Consistent markings

Consistent markings

A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

Consistent markings

A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

element $h \in H$ $\sim$ type of structures

Consistent markings

A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,. . .

element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures

Consistent markings

A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures
homomorphism $\alpha\colon \mathrm{Struct} \to H$	$\sim$	assignment of actual types

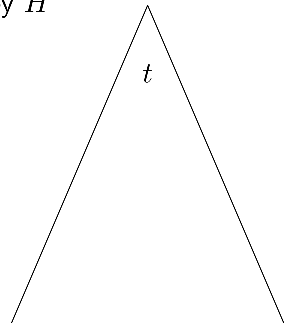
Consistent markings

A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures
homomorphism $\alpha\colon \mathrm{Struct} \to H$	$\sim$	assignment of actual types

Marking : a labelling γ of a partial tree t by H



Consistent markings

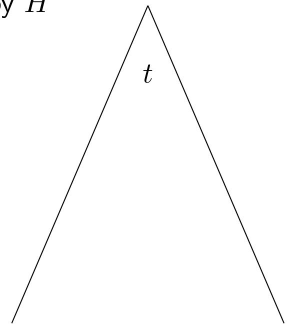
A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures
homomorphism $\alpha\colon \mathrm{Struct} \to H$	$\sim$	assignment of actual types

Marking : a labelling γ of a partial tree t by H

$$\gamma\colon \mathrm{dom}(t) \to H$$



Consistent markings

A finite algebra H

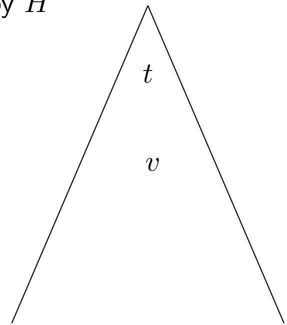
e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures
homomorphism $\alpha\colon \mathrm{Struct} \to H$	$\sim$	assignment of actual types

Marking : a labelling γ of a partial tree t by H

$$\gamma\colon \mathrm{dom}(t) \to H$$

$$\gamma(v) \equiv$$



Consistent markings

A finite algebra H

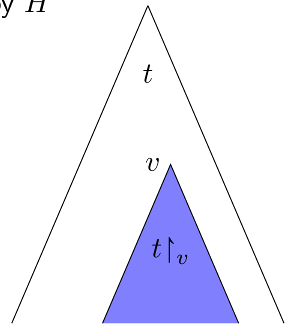
e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,. . .

element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures
homomorphism $\alpha\colon \mathrm{Struct} \to H$	$\sim$	assignment of actual types

Marking : a labelling γ of a partial tree t by H

$\gamma\colon \mathrm{dom}(t) \to H$

$\gamma(v) \equiv$ declared type of $t{\restriction}_v$



Consistent markings

A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

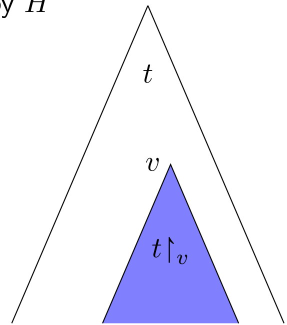
element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures
homomorphism $\alpha\colon \mathrm{Struct} \to H$	$\sim$	assignment of actual types

Marking : a labelling γ of a partial tree t by H

$$\gamma\colon \mathrm{dom}(t) \to H$$

$\gamma(v) \equiv$ declared type of $t\restriction_v$

Actual marking : $\gamma(v) = \alpha(t\restriction_v)$



Consistent markings

A finite algebra H

e.g. MSO-types Tp_k, monoid, forest algebra, thin algebra,...

element $h \in H$	$\sim$	type of structures
operation $\cdot$ in H	$\sim$	composition of structures
homomorphism $\alpha\colon \mathrm{Struct} \to H$	$\sim$	assignment of actual types

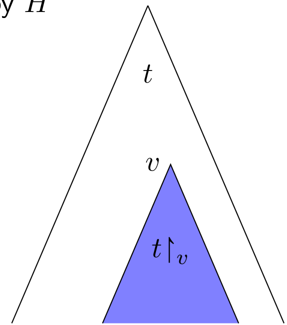
Marking : a labelling γ of a partial tree t by H

$$\gamma\colon \mathrm{dom}(t) \to H$$

$\gamma(v) \equiv$ declared type of $t{\upharpoonright}_v$

Actual marking : $\gamma(v) = \alpha(t{\upharpoonright}_v)$

if there exists $\alpha\colon \mathrm{Trees} \to H\ldots$



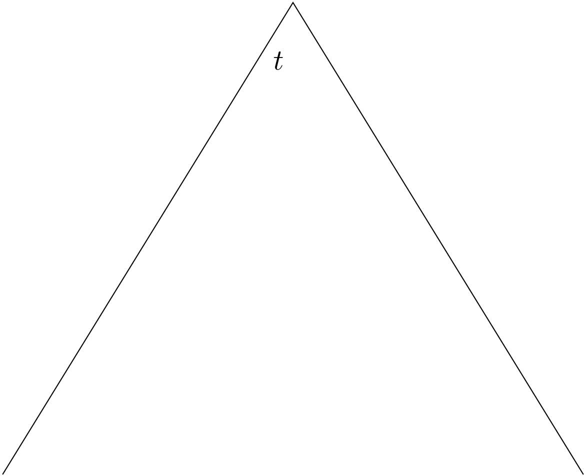
Consistent markings

Consistent markings

γ is consistent if the declarations are consistent along branches

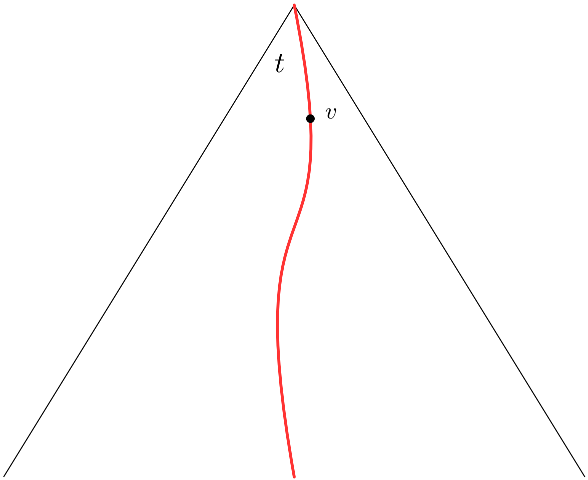
Consistent markings

γ is consistent if the declarations are consistent along branches



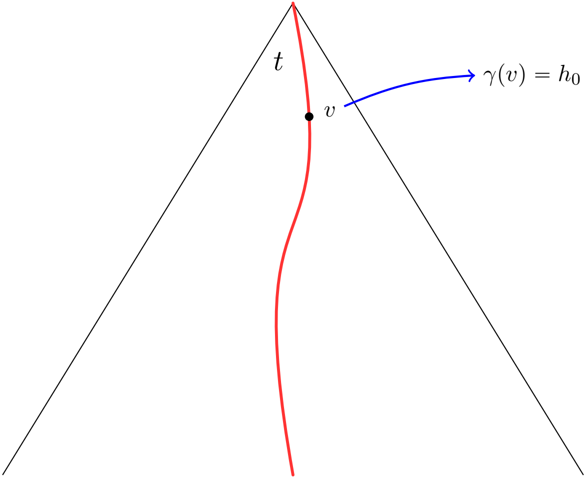
Consistent markings

γ is consistent if the declarations are consistent along branches



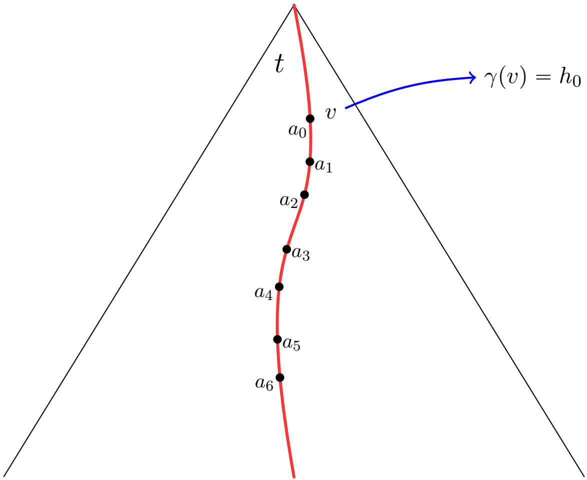
Consistent markings

γ is consistent if the declarations are consistent along branches



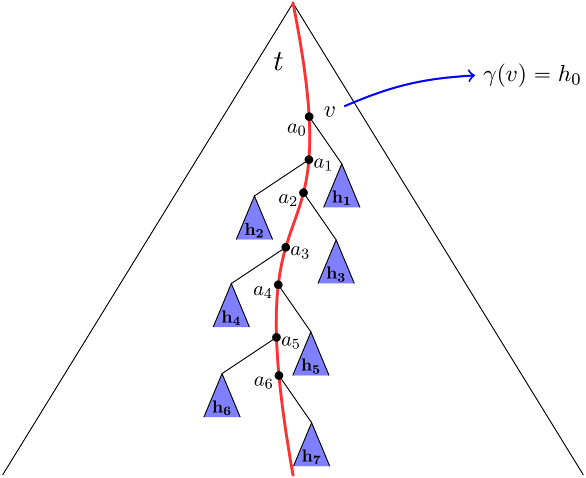
Consistent markings

γ is consistent if the declarations are consistent along branches



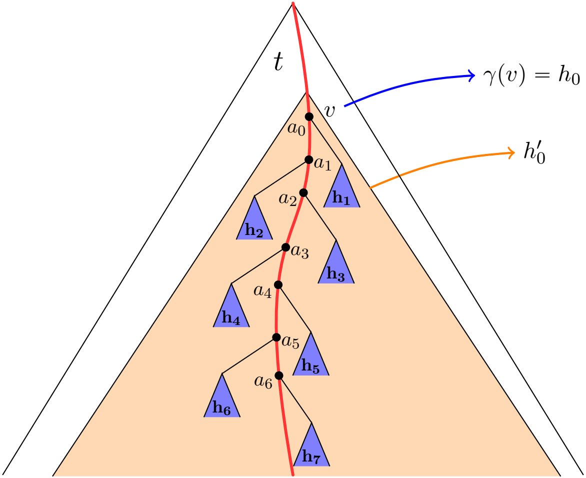
Consistent markings

γ is consistent if the declarations are consistent along branches



Consistent markings

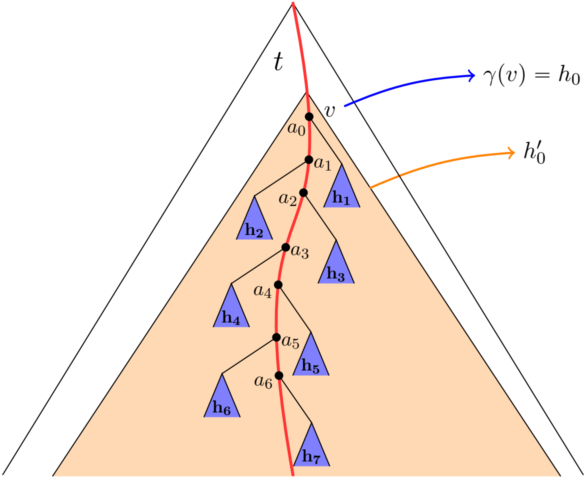
γ is consistent if the declarations are consistent along branches



Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h'_0$]



Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h'_0$]

Example

$$H = \{h_0, h_1\}, \quad h_0 \equiv \text{“no letter } a\text{”}, \quad h_1 \equiv \text{“exists letter } a\text{”}$$

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}, \quad h_0 \equiv$ "no letter a", $\quad h_1 \equiv$ "exists letter a"

For all v let:

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!

Consistent markings

γ is consistent if the declarations are consistent along branches

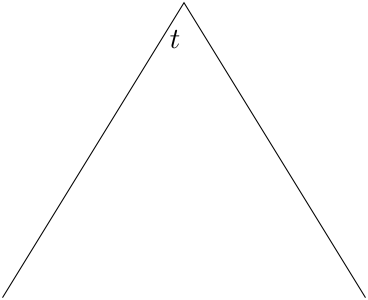
[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}, \quad h_0 \equiv$ "no letter a", $\quad h_1 \equiv$ "exists letter a"

For all v let: $\quad t(v) = b \quad$ and $\quad \gamma(v) = h_1$ ("exists a")

γ is consistent!



Consistent markings

γ is consistent if the declarations are consistent along branches

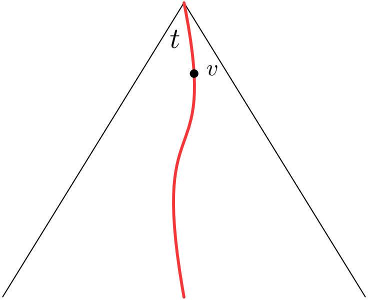
[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$$H = \{h_0, h_1\}, \quad h_0 \equiv \text{“no letter } a\text{”}, \quad h_1 \equiv \text{“exists letter } a\text{”}$$

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ (“exists a”)

γ is consistent!



Consistent markings

γ is consistent if the declarations are consistent along branches

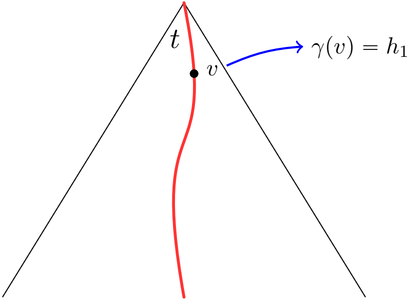
[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$$H = \{h_0, h_1\}, \quad h_0 \equiv \text{“no letter } a\text{”}, \quad h_1 \equiv \text{“exists letter } a\text{”}$$

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ (“exists a”)

γ is consistent!



Consistent markings

γ is consistent if the declarations are consistent along branches

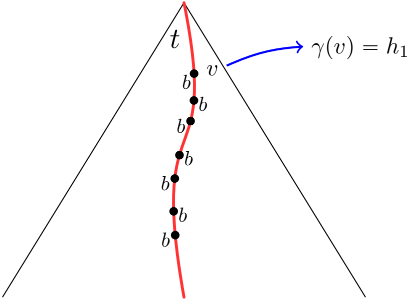
[it is enough to use thin algebra to check if $h_0 = h'_0$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!



Consistent markings

γ is consistent if the declarations are consistent along branches

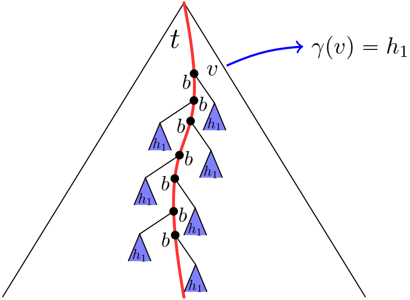
[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!



Consistent markings

γ is consistent if the declarations are consistent along branches

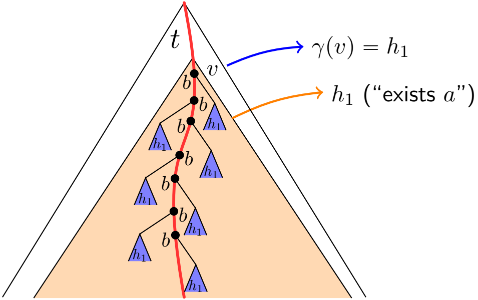
[it is enough to use thin algebra to check if $h_0 = h'_0$]

Example

$$H = \{h_0, h_1\}, \quad h_0 \equiv \text{"no letter } a\text{"}, \quad h_1 \equiv \text{"exists letter } a\text{"}$$

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!



Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h'_0$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}, \quad h_0 \equiv$ "no letter a", $\quad h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h_0'$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Consistent markings

γ is consistent if the declarations are consistent along branches

[it is enough to use thin algebra to check if $h_0 = h'_0$]

Example

$H = \{h_0, h_1\}$, $h_0 \equiv$ "no letter a", $h_1 \equiv$ "exists letter a"

For all v let: $t(v) = b$ and $\gamma(v) = h_1$ ("exists a")

γ is consistent!

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

[no actual marking because $\alpha\colon \mathrm{Scattered} \to H$ (not $\alpha\colon \mathrm{Trees} \to H$)]

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Proof

$(\Downarrow)$ Assume that H is a thin algebra without consistent marking.

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Proof

$(\Downarrow)$ Assume that H is a thin algebra without consistent marking.
Via determinacy we construct a choice function over thin trees.

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Proof

($\Downarrow$) Assume that H is a thin algebra without consistent marking.
Via determinacy we construct a choice function over thin trees.

($\Uparrow$) Assume that φ is an MSO-def. choice function over scattered trees.

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Proof

($\Downarrow$) Assume that H is a thin algebra without consistent marking.
Via determinacy we construct a choice function over thin trees.

($\Uparrow$) Assume that φ is an MSO-def. choice function over scattered trees.
Construct a thin algebra H that guides φ.

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Proof

($\Downarrow$) Assume that H is a thin algebra without consistent marking.
Via determinacy we construct a choice function over thin trees.

($\Uparrow$) Assume that φ is an MSO-def. choice function over scattered trees.
Construct a thin algebra H that guides φ.
Find a consistent marking $\gamma\colon \{0,1\}^{<\omega} \to H$.

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Proof

($\Downarrow$) Assume that H is a thin algebra without consistent marking.
Via determinacy we construct a choice function over thin trees.

($\Uparrow$) Assume that φ is an MSO-def. choice function over scattered trees.
Construct a thin algebra H that guides φ.
Find a consistent marking $\gamma\colon \{0,1\}^{<\omega} \to H$.
Guide φ using γ to find a *difficult* scattered tree t.

Theorem (S. [2013])

There is no MSO-def. choice function on thin trees

iff

For every finite thin algebra H and every tree t (scattered or not)

there exists a consistent marking of t by H.

Proof

($\Downarrow$) Assume that H is a thin algebra without consistent marking.
Via determinacy we construct a choice function over thin trees.

($\Uparrow$) Assume that φ is an MSO-def. choice function over scattered trees.
Construct a thin algebra H that guides φ.
Find a consistent marking $\gamma\colon \{0,1\}^{<\omega} \to H$.
Guide φ using γ to find a *difficult* scattered tree t.
$\rightsquigarrow$ φ is **not** a choice function on t.

Application : unambiguity

Application : unambiguity

$\mathcal{A}$ — non-deterministic tree automaton

Application : unambiguity $\mathcal{A}$ — non-deterministic tree automaton

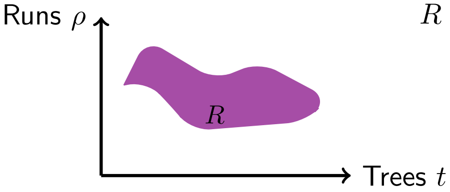
Application : unambiguity

$\mathcal{A}$ — non-deterministic tree automaton

$R = \{(t, \rho) \mid \rho$ is accepting over $t\}$

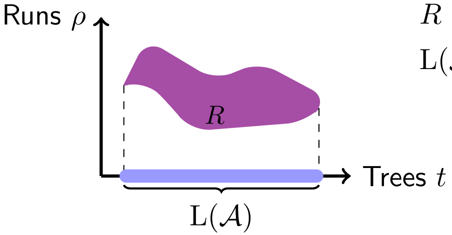
Application : unambiguity

$\mathcal{A}$ — non-deterministic tree automaton



$R = \{(t, \rho) \mid \rho$ is accepting over $t\}$

Application : unambiguity $\mathcal{A}$ — non-deterministic tree automaton

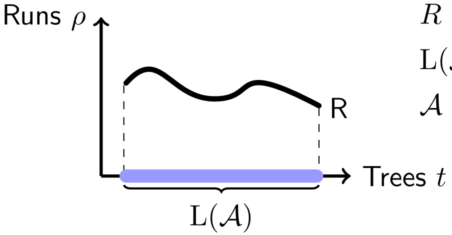


$R = \{(t, \rho) \mid \rho$ is accepting over $t\}$

$\mathrm{L}(\mathcal{A}) = \{t \mid \exists_\rho\ (t, \rho) \in R\}$

Application : unambiguity

$\mathcal{A}$ — non-deterministic tree automaton



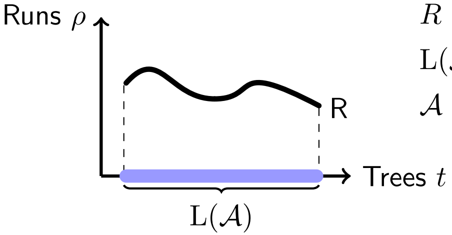
$R = \{(t, \rho) \mid \rho$ is accepting over $t\}$

$\mathrm{L}(\mathcal{A}) = \{t \mid \exists_\rho \ (t, \rho) \in R\}$

$\mathcal{A}$ is unambiguous if R is uniformised

Application : unambiguity

$\mathcal{A}$ — non-deterministic tree automaton



$R = \{(t, \rho) \mid \rho$ is accepting over $t\}$

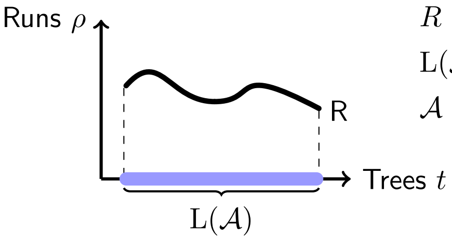
$\mathrm{L}(\mathcal{A}) = \{t \mid \exists_\rho\ (t, \rho) \in R\}$

$\mathcal{A}$ is unambiguous if R is uniformised

Theorem (Niwiński, Walukiewicz [1996])

The language $\exists_y\ a(y)$ cannot be recognised by any unambiguous automaton.

Application : unambiguity $\mathcal{A}$ — non-deterministic tree automaton



$R = \{(t, \rho) \mid \rho$ is accepting over $t\}$

$\mathrm{L}(\mathcal{A}) = \{t \mid \exists_\rho \ (t, \rho) \in R\}$

$\mathcal{A}$ is unambiguous if R is uniformised

Theorem (Niwiński, Walukiewicz [1996])

The language $\exists_y \ a(y)$ cannot be recognised by any unambiguous automaton.

Proof

Any unambiguous automaton for $\exists_y \ a(y)$ induces an MSO-definable choice function. ■

Bi-unambiguous languages

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}{-}L$ are unambiguous

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}{-}L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}-L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Problem

How to decide if L is bi-unambiguous?

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees} - L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Problem

How to decide if L is bi-unambiguous?

Theorem (Bilkowski, S. [2013])

If there is **no** MSO-def. choice function over scattered trees

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}-L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Problem

How to decide if L is bi-unambiguous?

Theorem (Bilkowski, S. [2013])

If there is **no** MSO-def. choice function over scattered trees

then it is decidable if a regular language of complete trees

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}-L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Problem

How to decide if L is bi-unambiguous?

Theorem (Bilkowski, S. [2013])

If there is **no** MSO-def. choice function over scattered trees
then it is decidable if a regular language of complete trees
is bi-unambiguous.

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}{-}L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Problem

How to decide if L is bi-unambiguous?

Theorem (Bilkowski, S. [2013])

If there is **no** MSO-def. choice function over scattered trees

then it is decidable if a regular language of complete trees

is bi-unambiguous.

[a concrete, sound algorithm $\mathcal{P}$ but. . .]

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}{-}L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Problem

How to decide if L is bi-unambiguous?

Theorem (Bilkowski, S. [2013])

If there is **no** MSO-def. choice function over scattered trees

then it is decidable if a regular language of complete trees

is bi-unambiguous.

$$\left[\begin{array}{c}\text{a concrete, sound algorithm } \mathcal{P} \text{ but...} \\ \text{completeness of } \mathcal{P} \text{ depends on Choice Conjecture}\end{array}\right]$$

Bi-unambiguous languages

L is bi-unambiguous if both L and $\mathrm{Trees}-L$ are unambiguous

$\rightsquigarrow$ boolean algebra of languages

Problem

How to decide if L is bi-unambiguous?

Theorem (Bilkowski, S. [2013])

If there is **no** MSO-def. choice function over scattered trees

then it is decidable if a regular language of complete trees

is bi-unambiguous.

$$\left[\begin{array}{c}\text{a concrete, sound algorithm } \mathcal{P} \text{ but...} \\ \text{completeness of } \mathcal{P} \text{ depends on Choice Conjecture}\end{array}\right]$$

Lemma (S. [2013])

If there is no MSO-def. choice function over scattered trees then

finite *prophetic* thin algebras are closed under homomorphisms.

Summary

Summary

- Uniformisability:

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]
- Choice:

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]
- Choice:
  - — MSO over finite/infinite words ✓

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]
- Choice:
  - — MSO over finite/infinite words ✓
  - — MSO over complete trees ✗ [GS83], [CL07]

Summary

- Uniformisability:

— FO over finite words	✗	(?)
— MSO over infinite words	✓	[S75], [R07]
— MSO over complete trees (with parameters)	✗	[GS83], [CL07]
— MSO over scattered trees (with parameters)	✓	[LS98], [BS13]
— MSO over scattered trees (**without** parameters)	✗	[BS13]

- Choice:

— MSO over finite/infinite words	✓	
— MSO over complete trees	✗	[GS83], [CL07]
— MSO over scattered trees	**???**	[BS13]

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]
- Choice:
  - — MSO over finite/infinite words ✓
  - — MSO over complete trees ✗ [GS83], [CL07]
  - — MSO over scattered trees **???** [BS13]
- Applications:

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]
- Choice:
  - — MSO over finite/infinite words ✓
  - — MSO over complete trees ✗ [GS83], [CL07]
  - — MSO over scattered trees **???** [BS13]
- Applications:
  - — thin algebras

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]
- Choice:
  - — MSO over finite/infinite words ✓
  - — MSO over complete trees ✗ [GS83], [CL07]
  - — MSO over scattered trees **???** [BS13]
- Applications:
  - — thin algebras
  - — bi-unambiguous languages of complete trees

Summary

- Uniformisability:
  - — FO over finite words ✗ (?)
  - — MSO over infinite words ✓ [S75], [R07]
  - — MSO over complete trees (with parameters) ✗ [GS83], [CL07]
  - — MSO over scattered trees (with parameters) ✓ [LS98], [BS13]
  - — MSO over scattered trees (**without** parameters) ✗ [BS13]
- Choice:
  - — MSO over finite/infinite words ✓
  - — MSO over complete trees ✗ [GS83], [CL07]
  - — MSO over scattered trees **???** [BS13]
- Applications:
  - — thin algebras
  - — bi-unambiguous languages of complete trees
  - — maybe parity index bounds for unambiguous languages. . .