Deciding the topological complexity of Büchi languages

Michał Skrzypczak   Igor Walukiewicz

ICALP 2016

Rome

Logics

Logics

Monadic Second-Order (MSO) logic:

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists^{\mathsf{fin}}_X$ X — finite set of nodes

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists^{\mathsf{fin}}_X$ X — finite set of nodes
- $\exists_X$ X — any set of nodes

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists^{\mathsf{fin}}_X$ X — finite set of nodes
- $\exists_X$ X — any set of nodes
- $x \in X$, $x = y$

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists^{\mathsf{fin}}_X$ X — finite set of nodes
- $\exists_X$ X — any set of nodes
- $x \in X$, $x = y$

Weak Monadic Second-Order (WMSO) logic:

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists^{\mathsf{fin}}_X$ X — finite set of nodes
- $\exists_X$ X — any set of nodes
- $x \in X$, $x = y$

Weak Monadic Second-Order (WMSO) logic:

- only $\exists_x$ and $\exists^{\mathsf{fin}}_X$ quantifiers

Logics

Monadic Second-Order ($\textsc{mso}$) logic:

- $\vee, \wedge, \neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists_X^{\mathsf{fin}}$ X — finite set of nodes
- $\exists_X$ X — any set of nodes
- $x \in X$, $x = y$

Weak Monadic Second-Order ($\textsc{wmso}$) logic:

- only $\exists_x$ and $\exists_X^{\mathsf{fin}}$ quantifiers

Existential Monadic Second-Order ($\exists\textsc{mso}$) logic:

Logics

Monadic Second-Order ($\textsc{mso}$) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists^{\mathsf{fin}}_X$ X — finite set of nodes
- $\exists_X$ X — any set of nodes
- $x \in X$, $x = y$

Weak Monadic Second-Order ($\textsc{wmso}$) logic:

- only $\exists_x$ and $\exists^{\mathsf{fin}}_X$ quantifiers

Existential Monadic Second-Order ($\exists\textsc{mso}$) logic:

- $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in \textsc{wmso}$

Logics

Monadic Second-Order (MSO) logic:

- $\vee$, $\wedge$, $\neg$ (Boolean connectives)
- $\exists_x$ x — node
- $\exists_X^{\mathsf{fin}}$ X — finite set of nodes
- $\exists_X$ X — any set of nodes
- $x \in X$, $x = y$

Weak Monadic Second-Order (WMSO) logic:

- only $\exists_x$ and $\exists_X^{\mathsf{fin}}$ quantifiers

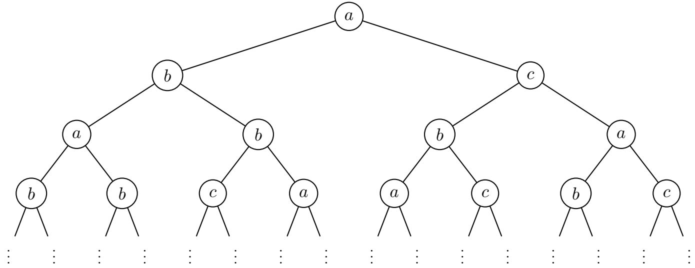
Existential Monadic Second-Order ($\exists$MSO) logic:

- $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO

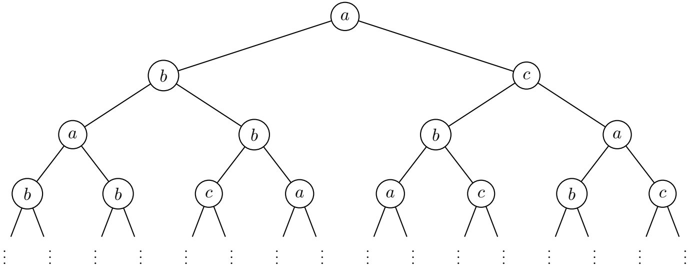
Other formalisms: LTL, CTL*, modal μ-calculus, ...

Decidability: infinite trees

Decidability: infinite trees

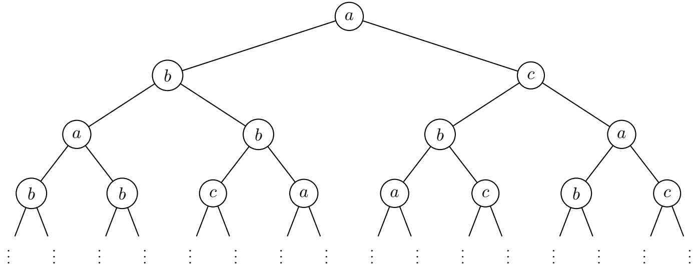


Decidability: infinite trees



$$t\colon \{\mathtt{L},\mathtt{R}\}^* \to A$$

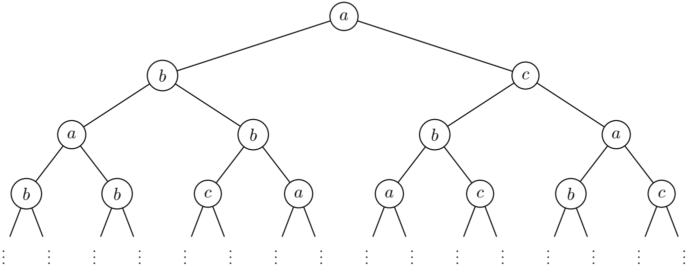
Decidability: infinite trees



$$t\colon \{\mathtt{L}, \mathtt{R}\}^* \to A$$

$$t \in A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

Decidability: infinite trees



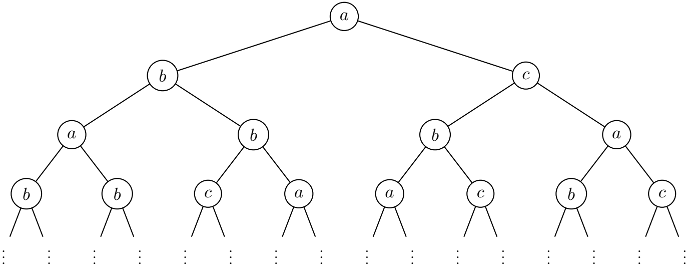
$$t\colon \{\mathtt{L}, \mathtt{R}\}^* \to A$$

$$t \in A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

Theorem (Rabin 1969)

The MSO theory of $\big(\{\mathtt{L}, \mathtt{R}\}^*, \mathrm{s}_{\mathtt{L}}, \mathrm{s}_{\mathtt{R}}\big)$ is decidable.

Decidability: infinite trees



$$t \colon \{\mathtt{L}, \mathtt{R}\}^* \to A$$

$$t \in A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

Theorem (Rabin 1969)

The MSO theory of $\big(\{\mathtt{L}, \mathtt{R}\}^*, \mathrm{s}_{\mathtt{L}}, \mathrm{s}_{\mathtt{R}}\big)$ is decidable.

☺ **"the mother of all decidability results"** ☺

Expressive power

Expressive power

Logics:

Expressive power

Logics:

WMSO

Expressive power

Logics: Quantifiers:

WMSO $\exists_x$, $\exists^{\mathsf{fin}}_X$

Expressive power

Logics:	Quantifiers:
WMSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO

Expressive power

Logics:	Quantifiers:
WMSO	$\exists_x,\ \exists^{\mathsf{fin}}_X$
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO
MSO	$\exists_x,\ \exists^{\mathsf{fin}}_X,\ \exists_X$

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$	
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	
MSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$, $\exists_X$	

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$	weak alternating
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	
MSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$, $\exists_X$	

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$	weak alternating
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
MSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$, $\exists_X$	

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$	weak alternating
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
MSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$, $\exists_X$	non-deterministic parity alternating parity

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$	weak alternating
$\rotatebox{90}{$\subseteq$}$		
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
$\rotatebox{90}{$\subseteq$}$		
MSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$, $\exists_X$	non-deterministic parity alternating parity

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$	weak alternating
$\cup\!\!\mid$		
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
$\cup\!\!\mid$		
MSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$, $\exists_X$	non-deterministic parity alternating parity

Question: are these containments strict?

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$	weak alternating
$\rotatebox{90}{$\subseteq$}$		
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
$\rotatebox{90}{$\subseteq$}$		
MSO	$\exists_x$, $\exists_X^{\mathsf{fin}}$, $\exists_X$	non-deterministic parity alternating parity

Question: are these containments strict? (yes)

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$	weak alternating
$\not\supseteq$		
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
$\not\supseteq$		
MSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$, $\exists_X$	non-deterministic parity alternating parity

Question: are these containments strict? (yes)

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x,\ \exists^{\mathsf{fin}}_X$	weak alternating
$\subsetneq$		
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
$\subsetneq$		
MSO	$\exists_x,\ \exists^{\mathsf{fin}}_X,\ \exists_X$	non-deterministic parity alternating parity

Question: are these containments strict? (yes)

Problem: characterise these logics

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x,\ \exists^{\mathsf{fin}}_X$	weak alternating
$\subsetneq$		
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
$\subsetneq$		
MSO	$\exists_x,\ \exists^{\mathsf{fin}}_X,\ \exists_X$	non-deterministic parity alternating parity

Question: are these containments strict? (yes)

Problem: characterise these logics

Input: φ from a stronger logic $\mathcal{L}$

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$	weak alternating
$\subsetneq$		
$\exists\mathrm{MSO}$	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in \mathrm{WMSO}$	non-deterministic Büchi alternating Büchi
$\subsetneq$		
MSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$, $\exists_X$	non-deterministic parity alternating parity

Question: are these containments strict? (yes)

Problem: characterise these logics

Input: φ from a stronger logic $\mathcal{L}$

Output: Yes if there is a formula equivalent to φ in a weaker logic $\mathcal{L}'$

Expressive power

Logics:	Quantifiers:	Automata:
WMSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$	weak alternating
$\not\supseteq$		
$\exists$MSO	$\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	non-deterministic Büchi alternating Büchi
$\not\supseteq$		
MSO	$\exists_x$, $\exists^{\mathsf{fin}}_X$, $\exists_X$	non-deterministic parity alternating parity

Question: are these containments strict? (yes)

Problem: characterise these logics (effective characterisation)

Input: φ from a stronger logic $\mathcal{L}$

Output: Yes if there is a formula equivalent to φ in a weaker logic $\mathcal{L}'$

Effective characterisations and expressive power

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, . . .

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])
It is decidable if a given $\exists$MSO formula is equivalent to a WMSO formula.

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, . . .

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])

It is decidable if a given $\exists$MSO formula is equivalent to a WMSO formula.

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])

It is decidable if a given $\exists$MSO formula is equivalent to a WMSO formula.

Proof

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])

It is decidable if a given $\exists$MSO formula is equivalent to a WMSO formula.

Proof

Theory of regular cost functions

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])

It is decidable if a given $\exists$MSO formula is equivalent to a WMSO formula.

Proof

Theory of regular cost functions

+

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])
It is decidable if a given ∃MSO formula is equivalent to a WMSO formula.

Proof

Theory of regular cost functions

+

reduction of Colcombet and Löding to

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, ...

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])

It is decidable if a given $\exists$MSO formula is equivalent to a WMSO formula.

Proof

Theory of regular cost functions

+

reduction of Colcombet and Löding to a boundedness problem

Effective characterisations and **expressive power**

Schutzenberger, McNaughton, Papert,

Thomas, Wilke, Kupferman, Vardi,

Place, Zeitoun, . . .

(mostly about words)

For infinite trees: $\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$

Theorem (Colcombet, Kuperberg, Löding, Vanden Boom [2013])

It is decidable if a given $\exists$MSO formula is equivalent to a WMSO formula.

Proof

Theory of regular cost functions

+

reduction of Colcombet and Löding to
a boundedness problem

⇝ not much insight about WMSO

Topology of infinite trees

Topology of infinite trees

$$A^{\left(\{\mathtt{L},\mathtt{R}\}^*\right)}$$

Topology of infinite trees

$$A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

$$\cong$$

Topology of infinite trees

$$A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

$$\cong$$

$$\{0,1\}^\omega$$

Topology of infinite trees

$$A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

$$\cong$$

$$\{0,1\}^\omega$$

$$\cong$$

Topology of infinite trees

$$A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

$$\cong$$

$$\{0,1\}^\omega$$

$$\cong$$

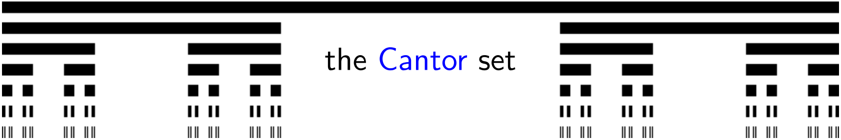
Topology of infinite trees

$$A^{(\{\mathtt{L},\mathtt{R}\}^*)}$$

$$\cong$$

$$\{0,1\}^\omega$$

$$\cong$$



the Cantor set

Descriptive set theory

Descriptive set theory

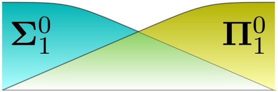
Start from simple sets

— open ($\boldsymbol{\Sigma}^0_1$) and closed ($\boldsymbol{\Pi}^0_1$)

Descriptive set theory

Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)



Descriptive set theory

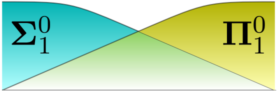
Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$



Descriptive set theory

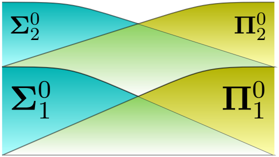
Start from simple sets

— open ($\boldsymbol{\Sigma}^0_1$) and closed ($\boldsymbol{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\boldsymbol{\Sigma}^0_{\eta+1}$ and $\boldsymbol{\Pi}^0_{\eta+1}$



Descriptive set theory

Start from simple sets

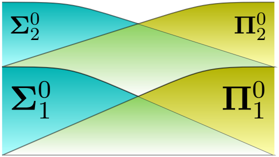
— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction



Descriptive set theory

Start from simple sets

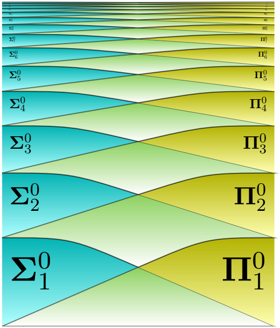
— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction



Descriptive set theory

Start from simple sets

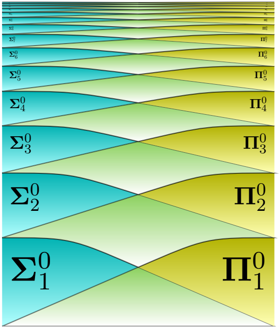
— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction (transfinite)



Descriptive set theory

Start from simple sets

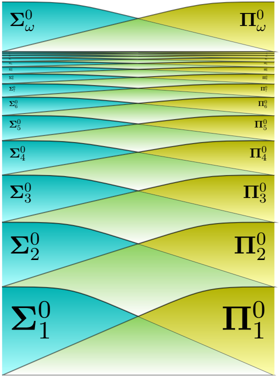
— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction (transfinite)



Descriptive set theory

Start from simple sets

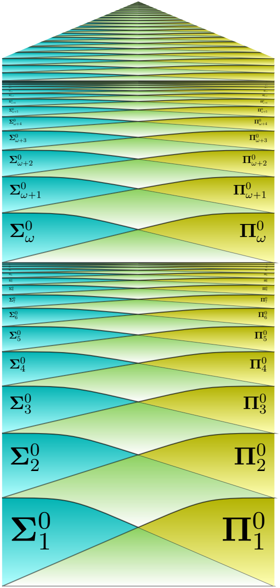
— open ($\boldsymbol{\Sigma}^0_1$) and closed ($\boldsymbol{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\boldsymbol{\Sigma}^0_{\eta+1}$ and $\boldsymbol{\Pi}^0_{\eta+1}$

By induction (transfinite)



Descriptive set theory

Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

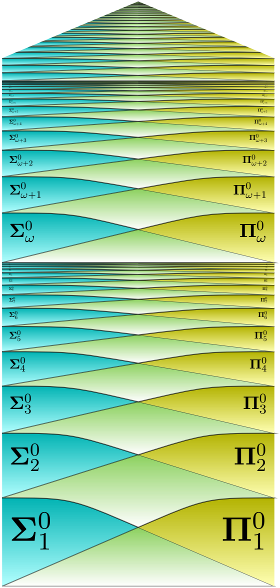
Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction (transfinite)

— Borel sets: $\mathbf{\Sigma}^0_\eta$, $\mathbf{\Pi}^0_\eta$ for $\eta < \omega_1$



Descriptive set theory

Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

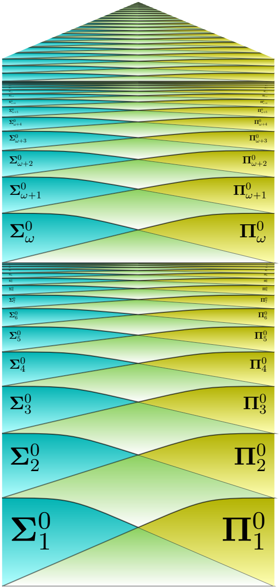
— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction (transfinite)

— Borel sets: $\mathbf{\Sigma}^0_\eta$, $\mathbf{\Pi}^0_\eta$ for $\eta < \omega_1$

Apply projection and co-projection

— analytic ($\mathbf{\Sigma}^1_1$) and co-analytic ($\mathbf{\Pi}^1_1$)



Descriptive set theory

Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

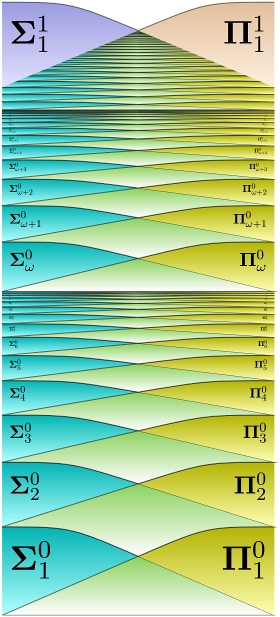
— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction (transfinite)

— Borel sets: $\mathbf{\Sigma}^0_\eta$, $\mathbf{\Pi}^0_\eta$ for $\eta < \omega_1$

Apply projection and co-projection

— analytic ($\mathbf{\Sigma}^1_1$) and co-analytic ($\mathbf{\Pi}^1_1$)



Descriptive set theory

Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

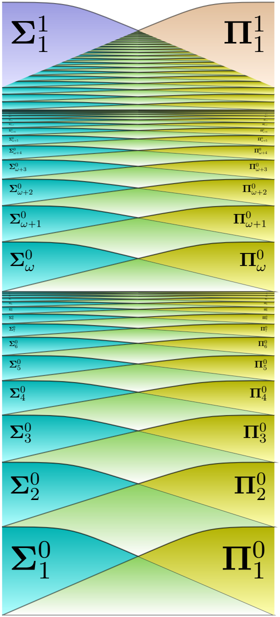
By induction (transfinite)

— Borel sets: $\mathbf{\Sigma}^0_\eta$, $\mathbf{\Pi}^0_\eta$ for $\eta < \omega_1$

Apply projection and co-projection

— analytic ($\mathbf{\Sigma}^1_1$) and co-analytic ($\mathbf{\Pi}^1_1$)

By induction



Descriptive set theory

Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

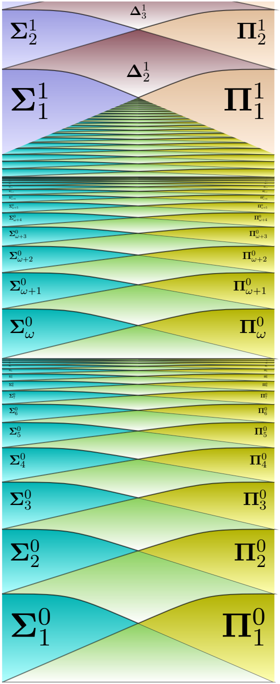
By induction (transfinite)

— Borel sets: $\mathbf{\Sigma}^0_\eta$, $\mathbf{\Pi}^0_\eta$ for $\eta < \omega_1$

Apply projection and co-projection

— analytic ($\mathbf{\Sigma}^1_1$) and co-analytic ($\mathbf{\Pi}^1_1$)

By induction



Descriptive set theory

Start from simple sets

— open ($\mathbf{\Sigma}^0_1$) and closed ($\mathbf{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

— $\mathbf{\Sigma}^0_{\eta+1}$ and $\mathbf{\Pi}^0_{\eta+1}$

By induction (transfinite)

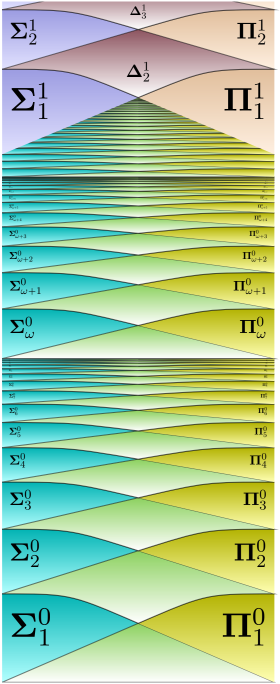
— Borel sets: $\mathbf{\Sigma}^0_\eta$, $\mathbf{\Pi}^0_\eta$ for $\eta < \omega_1$

Apply projection and co-projection

— analytic ($\mathbf{\Sigma}^1_1$) and co-analytic ($\mathbf{\Pi}^1_1$)

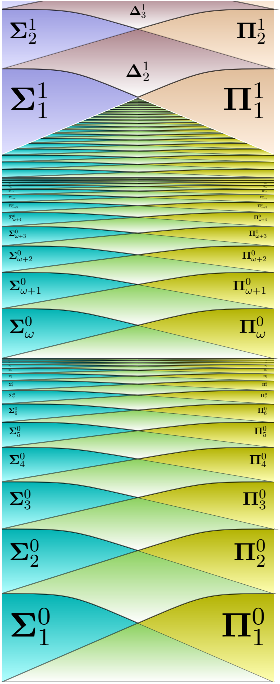
By induction

— projective sets: $\mathbf{\Sigma}^1_n$, $\mathbf{\Pi}^1_n$ for $n < \omega$

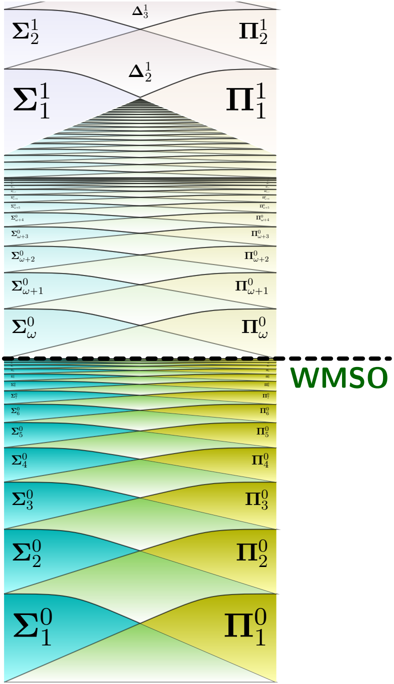


Upper bounds

Upper bounds

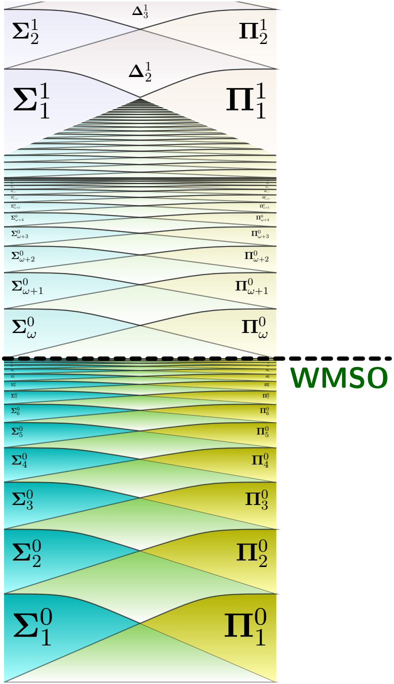


Upper bounds



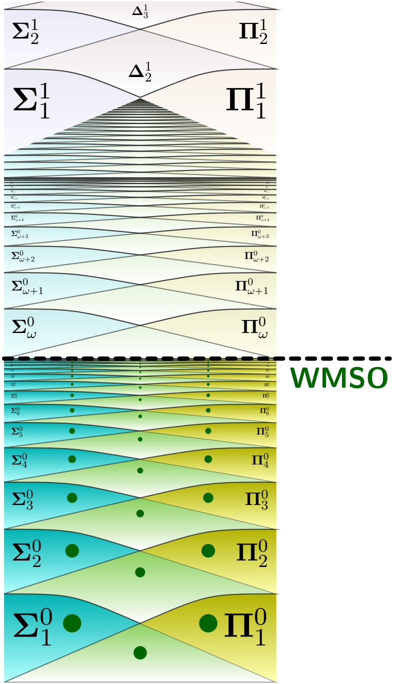
Upper bounds

$$L \in \mathrm{WMSO} \implies L \in \boldsymbol{\Sigma}^0_n \ \text{(for some } n\text{)}$$



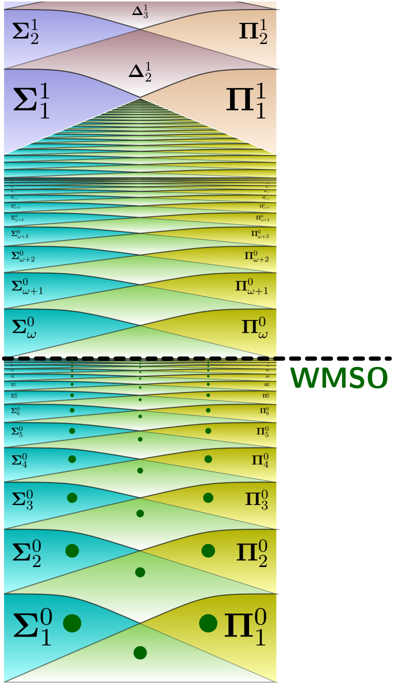
Upper bounds

$L \in \mathrm{WMSO} \implies L \in \mathbf{\Sigma}^0_n$ (for some n)



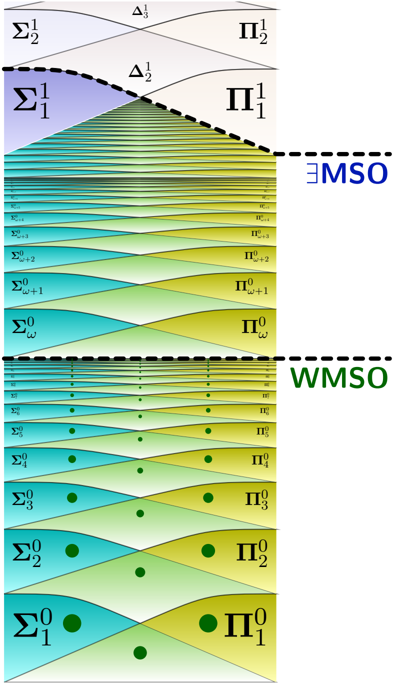
Upper bounds

$$L \in \text{WMSO} \implies L \in \boldsymbol{\Sigma}^0_n \quad \text{(for some } n\text{)}$$



Upper bounds

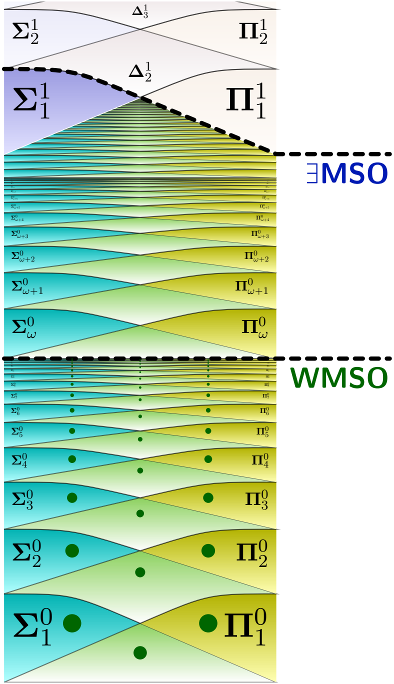
$$L \in \text{WMSO} \implies L \in \mathbf{\Sigma}^0_n \text{ (for some } n)$$



Upper bounds

$L \in \mathrm{WMSO} \implies L \in \mathbf{\Sigma}^0_n$ (for some n)

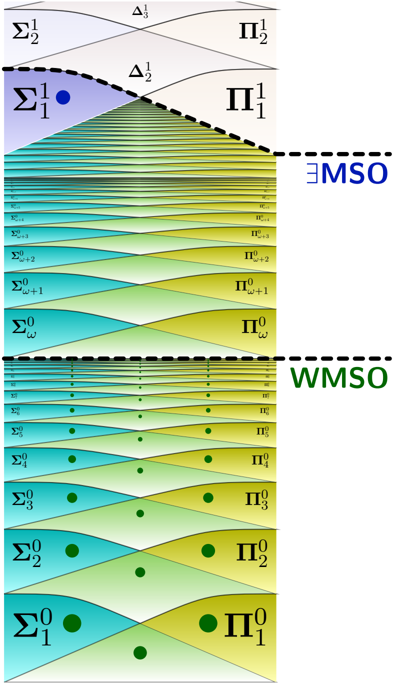
$L \in \exists\mathrm{MSO} \implies L \in \mathbf{\Sigma}^1_1$



Upper bounds

$$L \in \text{WMSO} \implies L \in \mathbf{\Sigma}^0_n \quad \text{(for some } n\text{)}$$

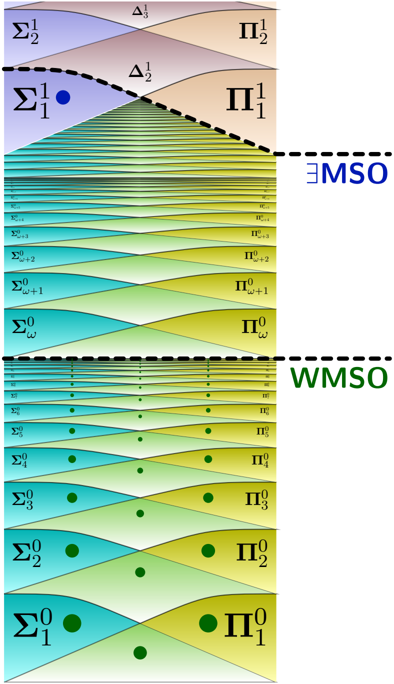
$$L \in \exists\text{MSO} \implies L \in \mathbf{\Sigma}^1_1$$



Upper bounds

$$L \in \text{WMSO} \implies L \in \mathbf{\Sigma}^0_n \quad \text{(for some } n\text{)}$$

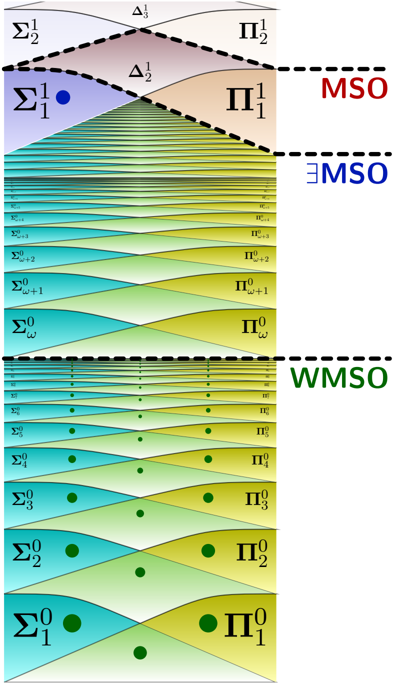
$$L \in \exists\text{MSO} \implies L \in \mathbf{\Sigma}^1_1$$



Upper bounds

$$L \in \text{WMSO} \implies L \in \mathbf{\Sigma}^0_n \quad \text{(for some } n\text{)}$$

$$L \in \exists\text{MSO} \implies L \in \mathbf{\Sigma}^1_1$$

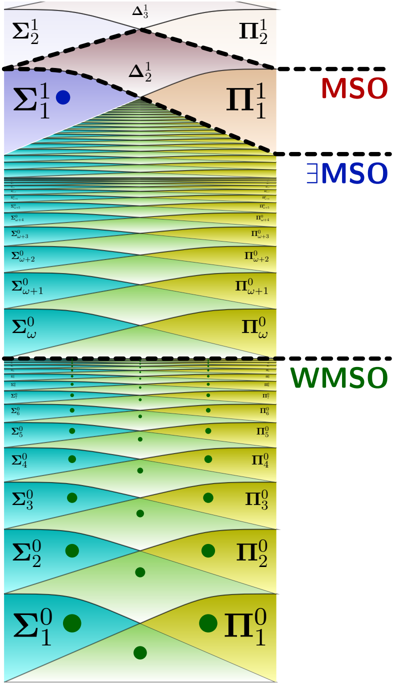


Upper bounds

$$L \in \text{WMSO} \implies L \in \mathbf{\Sigma}^0_n \quad \text{(for some } n\text{)}$$

$$L \in \exists\text{MSO} \implies L \in \mathbf{\Sigma}^1_1$$

$$L \in \text{MSO} \implies L \in \mathbf{\Delta}^1_2$$

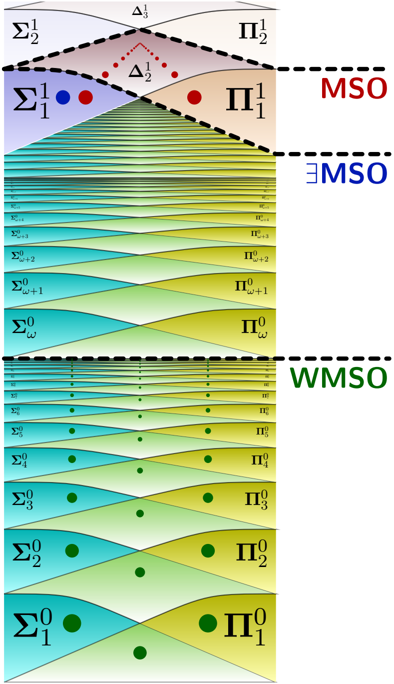


Upper bounds

$L \in \text{WMSO} \implies L \in \mathbf{\Sigma}^0_n$ (for some n)

$L \in \exists\text{MSO} \implies L \in \mathbf{\Sigma}^1_1$

$L \in \text{MSO} \implies L \in \mathbf{\Delta}^1_2$

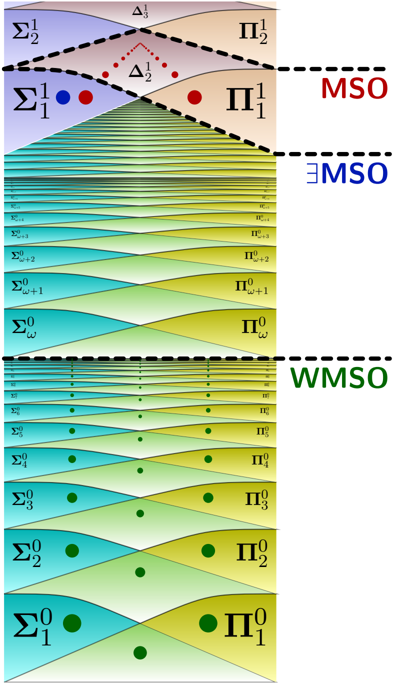


Upper bounds

$L \in \mathrm{WMSO} \implies L \in \mathbf{\Sigma}^0_n$ (for some n)

$L \in \exists\mathrm{MSO} \implies L \in \mathbf{\Sigma}^1_1$

$L \in \mathrm{MSO} \implies L \in \mathbf{\Delta}^1_2$



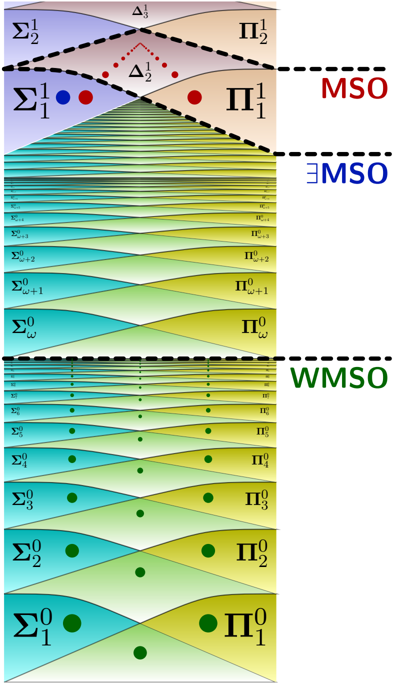
Upper bounds

$$L \in \mathrm{WMSO} \implies L \in \mathbf{\Sigma}^0_n \quad \text{(for some } n\text{)}$$

$$L \in \exists\mathrm{MSO} \implies L \in \mathbf{\Sigma}^1_1$$

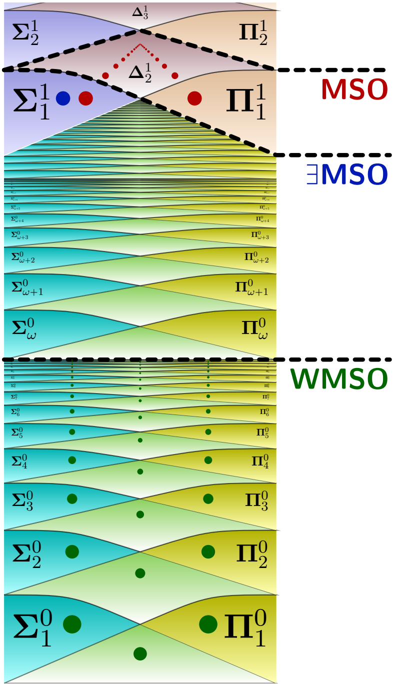
$$L \in \mathrm{MSO} \implies L \in \mathbf{\Delta}^1_2$$

$$\mathrm{WMSO} \subsetneq \exists\mathrm{MSO} \subsetneq \mathrm{MSO}$$

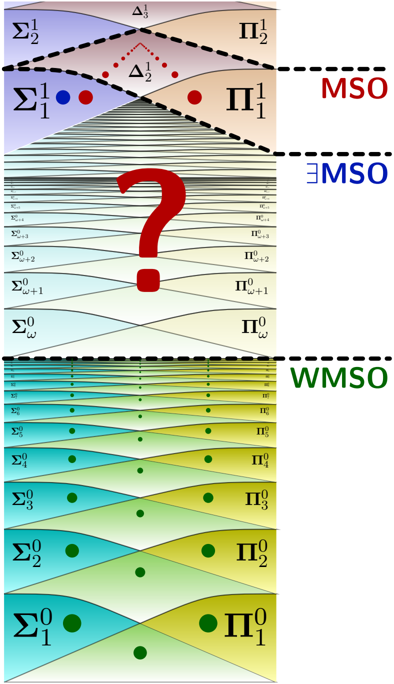


Gap property

Gap property



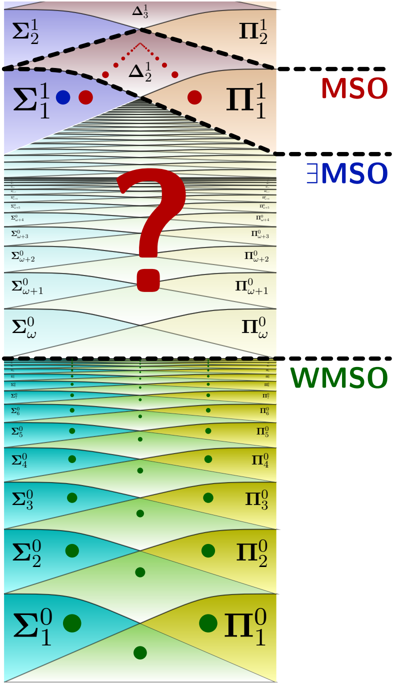
Gap property



Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language is WMSO-definable.

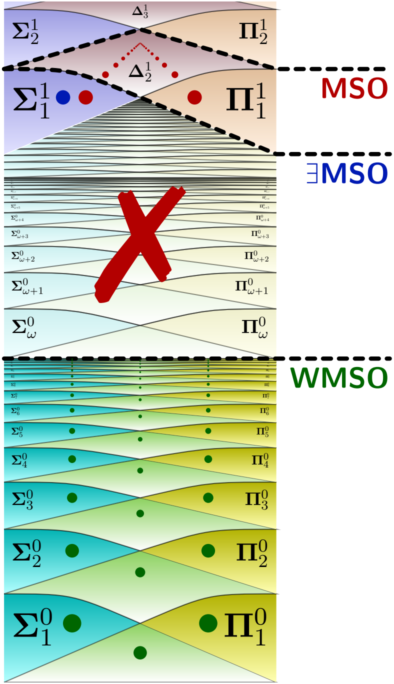


Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language

is WMSO-definable.



Gap property

Conjecture (Skurczyński [1993])

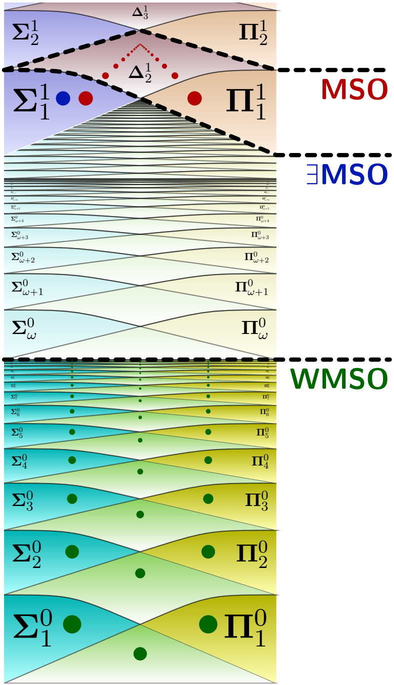
A Borel MSO-definable language is WMSO-definable.

Theorem (Walukiewicz, S. [2016])

An ∃MSO-definable language is either:

— WMSO-definable and Borel

— **not** WMSO-definable and $\boldsymbol{\Sigma}^1_1$-comp.



Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language

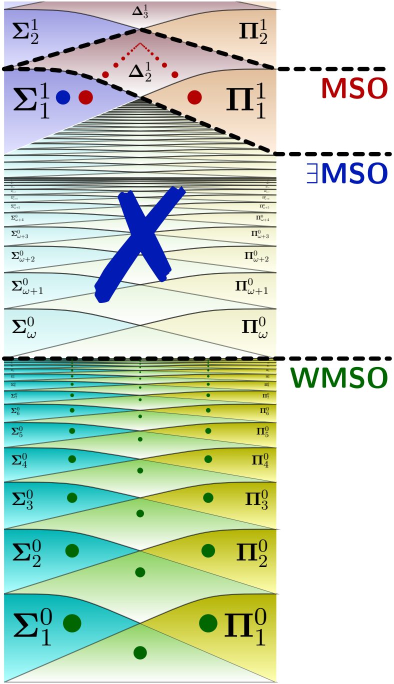
is WMSO-definable.

Theorem (Walukiewicz, S. [2016])

An $\exists$MSO-definable language is either:

— WMSO-definable and Borel

— **not** WMSO-definable and Σ^1_1-comp.



Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language is WMSO-definable.

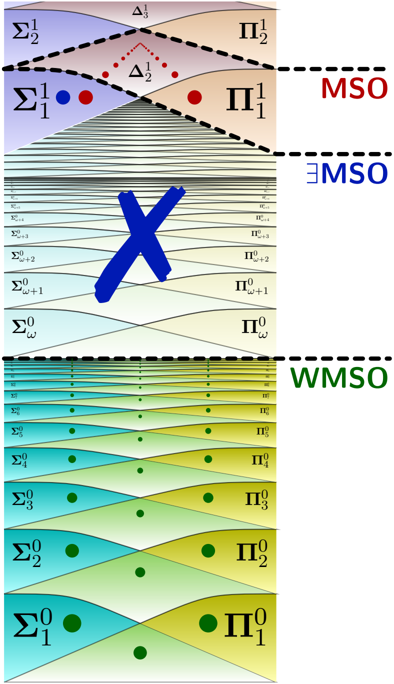
Theorem (Walukiewicz, S. [2016])

An ∃MSO-definable language is either:

— WMSO-definable and Borel

— **not** WMSO-definable and $\mathbf{\Sigma}^1_1$-comp.

And the dychotomy is decidable.



Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language

is WMSO-definable.

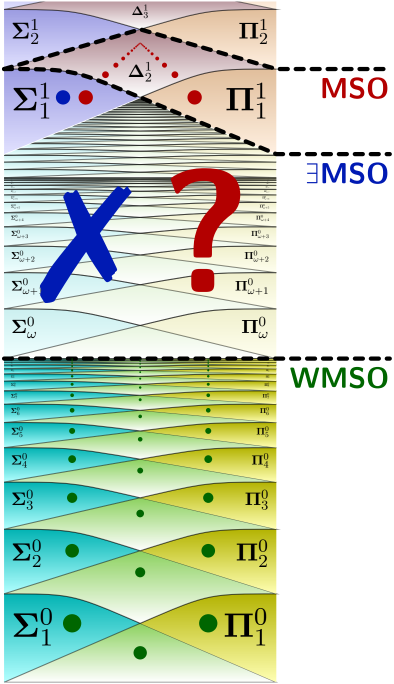
Theorem (Walukiewicz, S. [2016])

An $\exists$MSO-definable language is either:

— WMSO-definable and Borel

— **not** WMSO-definable and $\mathbf{\Sigma}^1_1$-comp.

And the dychotomy is decidable.



Theorem (Walukiewicz, S. [2016])

$L \in \exists\textsc{mso}$ is effectively either:

— $\textsc{wmso}$-definable

— $\boldsymbol{\Sigma}^1_1$-complete

Theorem (Walukiewicz, S. [2016])

$L \in \exists\textsc{mso}$ is effectively either:

— $\textsc{wmso}$-definable

— $\mathbf{\Sigma}_1^1$-complete

Proof

Theorem (Walukiewicz, S. [2016])

$L \in \exists\text{MSO}$ is effectively either:

— WMSO-definable

— $\boldsymbol{\Sigma}^1_1$-complete

Proof

Input $\varphi \in \exists\text{MSO}$ (or a Büchi automaton) for L

Theorem (Walukiewicz, S. [2016])

$L \in \exists\text{MSO}$ is effectively either:

— WMSO-definable

— $\mathbf{\Sigma}_1^1$-complete

Proof

Input $\varphi \in \exists\text{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

Theorem (Walukiewicz, S. [2016])

$L \in \exists\mathrm{MSO}$ is effectively either:

— WMSO-definable

— $\mathbf{\Sigma}_1^1$-complete

Proof

Input $\varphi \in \exists\mathrm{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

Theorem (Walukiewicz, S. [2016])

$L \in \exists\mathrm{MSO}$ is effectively either:

— WMSO-definable

— $\boldsymbol{\Sigma}^1_1$-complete

Proof

Input $\varphi \in \exists\mathrm{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

$\sigma_\forall$ — finite memory winning strategy

Theorem (Walukiewicz, S. [2016])

$L \in \exists\text{MSO}$ is effectively either:

— WMSO-definable

— $\mathbf{\Sigma}_1^1$-complete

Proof

Input $\varphi \in \exists\text{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

$\sigma_\forall$ — finite memory winning strategy

$\sigma_\forall$ gives a weak alternating automaton for L

Theorem (Walukiewicz, S. [2016])

$L \in \exists\text{MSO}$ is effectively either:

— WMSO-definable

— $\boldsymbol{\Sigma}_1^1$-complete

Proof

Input $\varphi \in \exists\text{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

$\sigma_\forall$ — finite memory winning strategy $\rightsquigarrow L \in \text{WMSO}$

$\sigma_\forall$ gives a weak alternating automaton for L

Theorem (Walukiewicz, S. [2016])

$L \in \exists\mathrm{MSO}$ is effectively either:

— WMSO-definable

— $\mathbf{\Sigma}^1_1$-complete

Proof

Input $\varphi \in \exists\mathrm{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

$\sigma_\forall$ — finite memory winning strategy $\rightsquigarrow L \in \mathrm{WMSO}$

$\sigma_\forall$ gives a weak alternating automaton for L

$\rightsquigarrow$ If $\exists$ wins $\mathcal{F}$:

Theorem (Walukiewicz, S. [2016])

$L \in \exists\text{MSO}$ is effectively either:

— WMSO-definable

— $\mathbf{\Sigma}_1^1$-complete

Proof

Input $\varphi \in \exists\text{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

$\sigma_\forall$ — finite memory winning strategy $\rightsquigarrow L \in \text{WMSO}$

$\sigma_\forall$ gives a weak alternating automaton for L

$\rightsquigarrow$ If $\exists$ wins $\mathcal{F}$:

$\sigma_\exists$ — a winning strategy

Theorem (Walukiewicz, S. [2016])

$L \in \exists\mathrm{MSO}$ is effectively either:

— WMSO-definable

— $\boldsymbol{\Sigma}^1_1$-complete

Proof

Input $\varphi \in \exists\mathrm{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

$\sigma_\forall$ — finite memory winning strategy $\rightsquigarrow$ $L \in \mathrm{WMSO}$

$\sigma_\forall$ gives a weak alternating automaton for L

$\rightsquigarrow$ If $\exists$ wins $\mathcal{F}$:

$\sigma_\exists$ — a winning strategy

$\sigma_\exists$ gives a continuous reduction of IF to L

Theorem (Walukiewicz, S. [2016])

$L \in \exists\mathrm{MSO}$ is effectively either:

— WMSO-definable

— $\boldsymbol{\Sigma}^1_1$-complete

Proof

Input $\varphi \in \exists\mathrm{MSO}$ (or a Büchi automaton) for L

Construct a finite game $\mathcal{F}$ with an ω-regular winning condition

$\rightsquigarrow$ If $\forall$ wins $\mathcal{F}$:

$\sigma_\forall$ — finite memory winning strategy $\rightsquigarrow$ $L \in \mathrm{WMSO}$

$\sigma_\forall$ gives a weak alternating automaton for L

$\rightsquigarrow$ If $\exists$ wins $\mathcal{F}$:

$\sigma_\exists$ — a winning strategy $\rightsquigarrow$ L is $\boldsymbol{\Sigma}^1_1$-comp.

$\sigma_\exists$ gives a continuous reduction of IF to L

Summary

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\mathrm{MSO}$

if $L \in \mathrm{WMSO}$.

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\mathrm{MSO}$

if $L \in \mathrm{WMSO}$.

Proof

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\mathrm{MSO}$

if $L \in \mathrm{WMSO}$.

Proof

Regular cost functions

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\text{MSO}$

if $L \in \text{WMSO}$.

Proof

Regular cost functions

+

Boundedness problem

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\text{MSO}$

if $L \in \text{WMSO}$.

Proof

Regular cost functions
+
Boundedness problem

Theorem (Walukiewicz, S. [2016])

$L \in \exists\text{MSO}$ is effectively either:

— WMSO-definable

— $\mathbf{\Sigma}^1_1$-complete

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\mathrm{MSO}$

if $L \in \mathrm{WMSO}$.

Proof

Regular cost functions

+

Boundedness problem

Theorem (Walukiewicz, S. [2016])

$L \in \exists\mathrm{MSO}$ is effectively either:

— WMSO-definable

— $\boldsymbol{\Sigma}_1^1$-complete

(direct constructions)

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\mathrm{MSO}$

if $L \in \mathrm{WMSO}$.

Proof

Regular cost functions

+

Boundedness problem

Theorem (Walukiewicz, S. [2016])

$L \in \exists\mathrm{MSO}$ is effectively either:

— WMSO-definable

— $\mathbf{\Sigma}^1_1$-complete

(direct constructions)

(EXPTIME algorithm)

Summary

Theorem (CKLV [2013])
It is decidable for $L \in \exists\text{MSO}$

if $L \in \text{WMSO}$.

Proof

Regular cost functions
+
Boundedness problem

Theorem (Walukiewicz, S. [2016])

$L \in \exists\text{MSO}$ is effectively either:

— WMSO-definable

— $\boldsymbol{\Sigma}^1_1$-complete

(direct constructions)

(EXPTIME algorithm)