Deciding the topological complexity of Büchi languages

Michał Skrzypczak **Igor Walukiewicz**

Highlights 2016
Brussels

Infinite trees

Infinite trees

Logic **Automata**

Infinite trees

Logic | **Automata**

Weak Monadic Second-Order (WMSO)

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

Infinite trees

Logic

Weak Monadic Second-Order (WMSO)

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

Automata

weak alternating

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists_X^{\text{fin}}$ quantifiers	(WMSO)	weak alternating
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n} \psi$ for $\psi \in$ WMSO	($\exists$MSO)	

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\text{fin}}_X$ quantifiers	(WMSO)	weak alternating
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	($\exists$MSO)	non-deterministic Büchi alternating Büchi

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\text{fin}}_X$ quantifiers	(WMSO)	weak alternating
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	($\exists$MSO)	non-deterministic Büchi alternating Büchi
Full Monadic Second-Order - $\exists_x$, $\exists^{\text{fin}}_X$, and $\exists_X$ quantifiers	(MSO)	

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists_X^{\text{fin}}$ quantifiers	(WMSO)	weak alternating
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	($\exists$MSO)	non-deterministic Büchi alternating Büchi
Full Monadic Second-Order - $\exists_x$, $\exists_X^{\text{fin}}$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

Infinite trees

Logic		**Automata**
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\text{fin}}_X$ quantifiers	(**WMSO**)	weak alternating
	$\subseteq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ **WMSO**	(**∃MSO**)	non-deterministic Büchi alternating Büchi
	$\subseteq$	
Full Monadic Second-Order - $\exists_x$, $\exists^{\text{fin}}_X$, and $\exists_X$ quantifiers	(**MSO**)	non-deterministic parity alternating parity

Infinite trees

Logic		**Automata**
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\mathsf{fin}}_X$ quantifiers	(WMSO)	weak alternating
	$\not\supseteq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	(∃MSO)	non-deterministic Büchi alternating Büchi
	$\not\supseteq$	
Full Monadic Second-Order - $\exists_x$, $\exists^{\mathsf{fin}}_X$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\mathsf{fin}}_X$ quantifiers	(WMSO)	weak alternating
	$\not\supseteq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	(∃MSO)	non-deterministic Büchi alternating Büchi
	$\not\supseteq$	
Full Monadic Second-Order - $\exists_x$, $\exists^{\mathsf{fin}}_X$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

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

It is decidable if a given ∃MSO formula is equivalent to a WMSO formula.

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\text{fin}}_X$ quantifiers	(WMSO)	weak alternating
	$\not\supseteq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	(∃MSO)	non-deterministic Büchi alternating Büchi
	$\not\supseteq$	
Full Monadic Second-Order - $\exists_x$, $\exists^{\text{fin}}_X$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

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

It is decidable if a given ∃MSO formula is equivalent to a WMSO formula.

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\mathsf{fin}}_X$ quantifiers	(WMSO)	weak alternating
	$\not\supsetneq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	(∃MSO)	non-deterministic Büchi alternating Büchi
	$\not\supsetneq$	
Full Monadic Second-Order - $\exists_x$, $\exists^{\mathsf{fin}}_X$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

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

It is decidable if a given ∃MSO formula is equivalent to a WMSO formula.

Proof

Infinite trees

Logic		**Automata**
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\text{fin}}_X$ quantifiers	(WMSO)	weak alternating
	$\not\supseteq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	($\exists$MSO)	non-deterministic Büchi alternating Büchi
	$\not\supseteq$	
Full Monadic Second-Order - $\exists_x$, $\exists^{\text{fin}}_X$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

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

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists^{\text{fin}}_X$ quantifiers	(WMSO)	weak alternating
	$\not\supseteq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	(∃MSO)	non-deterministic Büchi alternating Büchi
	$\not\supseteq$	
Full Monadic Second-Order - $\exists_x$, $\exists^{\text{fin}}_X$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

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

+

Infinite trees

Logic | **Automata**

Weak Monadic Second-Order (WMSO) — weak alternating

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

$\not\supseteq$

Existential Monadic Second-Order ($\exists$MSO) — non-deterministic Büchi, alternating Büchi

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

$\not\supseteq$

Full Monadic Second-Order (MSO) — non-deterministic parity, alternating parity

- $\exists_x$, $\exists_X^{\text{fin}}$, and $\exists_X$ quantifiers

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

Infinite trees

Logic		Automata
Weak Monadic Second-Order - only $\exists_x$ and $\exists_X^{\mathsf{fin}}$ quantifiers	(WMSO)	weak alternating
	$\subsetneq$	
Existential Monadic Second-Order - $\exists_{X_1} \ldots \exists_{X_n}\ \psi$ for $\psi \in$ WMSO	(∃MSO)	non-deterministic Büchi alternating Büchi
	$\subsetneq$	
Full Monadic Second-Order - $\exists_x$, $\exists_X^{\mathsf{fin}}$, and $\exists_X$ quantifiers	(MSO)	non-deterministic parity alternating parity

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
a boundedness problem

Topology of infinite trees

Topology of infinite trees

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

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

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

$$\Vert\wr$$

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

$$\Vert\wr$$

Topology of infinite trees

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

$$\cong$$

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

$$\cong$$

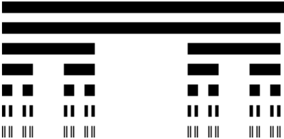
Topology of infinite trees

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

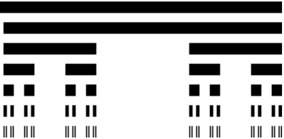
$$\cong$$

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

$$\cong$$



the Cantor set

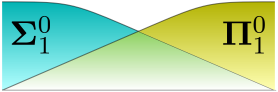


Descriptive set theory

Descriptive set theory

Start from simple sets

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



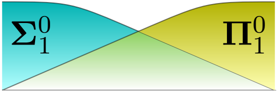
Descriptive set theory

Start from simple sets

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

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)



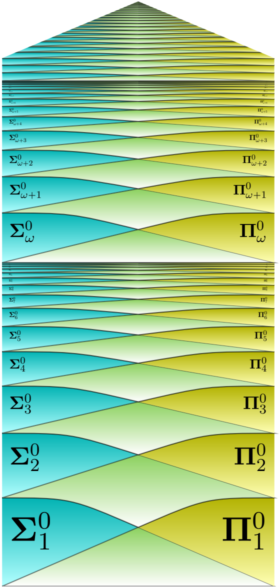
Descriptive set theory

Start from simple sets

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

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)



Descriptive set theory

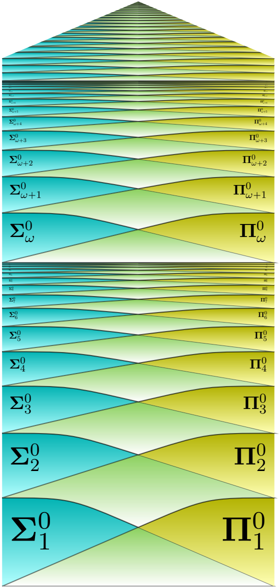
Start from simple sets

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

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

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



Descriptive set theory

Start from simple sets

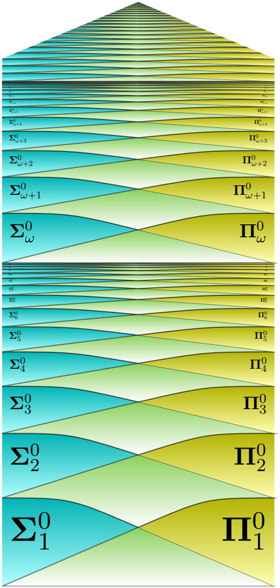
$\rightsquigarrow$ open ($\boldsymbol{\Sigma}^0_1$) and closed ($\boldsymbol{\Pi}^0_1$)

Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

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

Apply projection and co-projection



Descriptive set theory

Start from simple sets

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

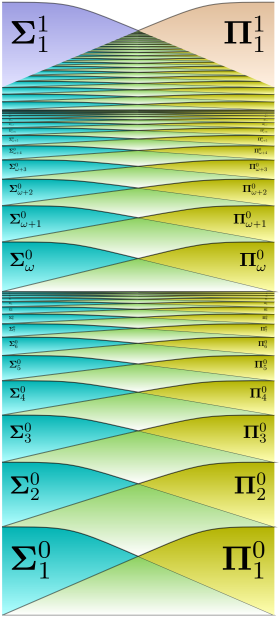
Apply countable unions ($\bigcup$)

and countable intersections ($\bigcap$)

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

Apply projection and co-projection

$\rightsquigarrow$ 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 ($\bigcup$)

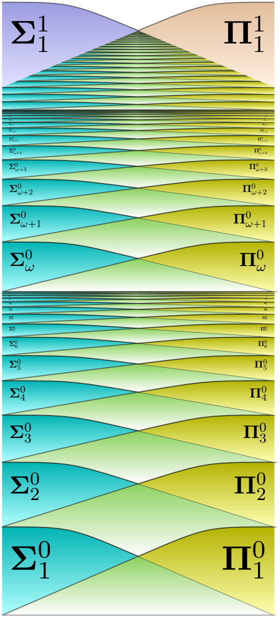
and countable intersections ($\bigcap$)

⇝ 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 ($\bigcap$)

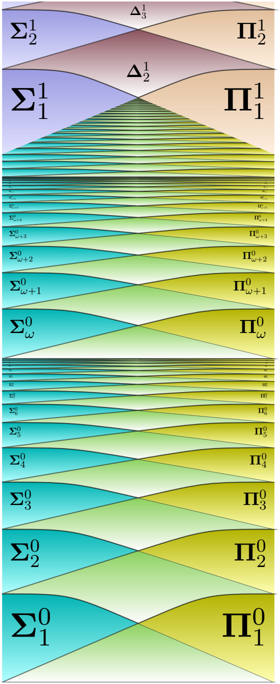
⇝ Borel sets: $\mathbf{\Sigma}^0_\eta$, $\mathbf{\Pi}^0_\eta$ for $\eta < \omega_1$

Apply projection and co-projection

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

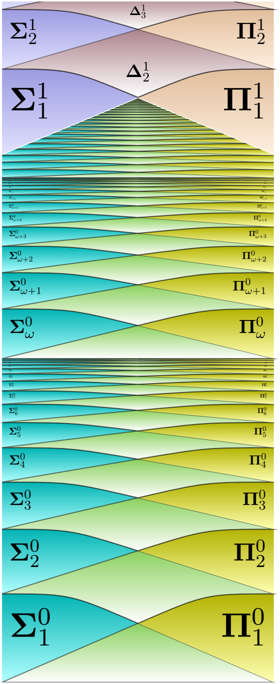
By induction

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



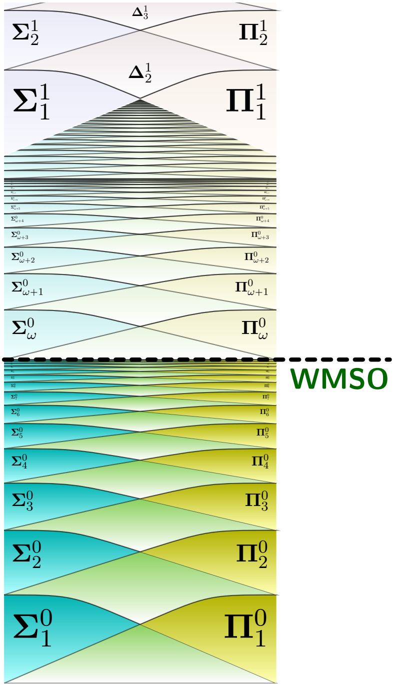
Upper bounds

Upper bounds



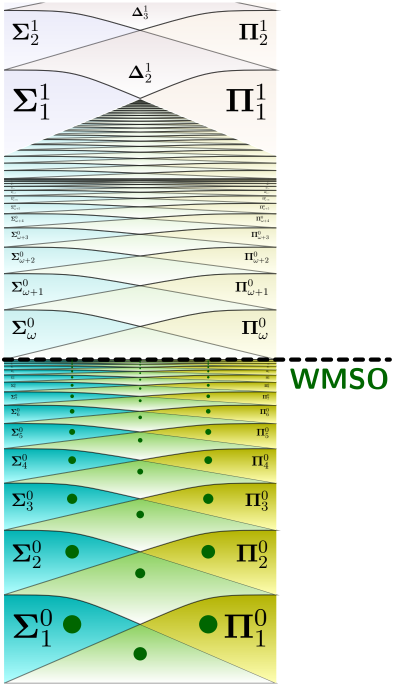
Upper bounds

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



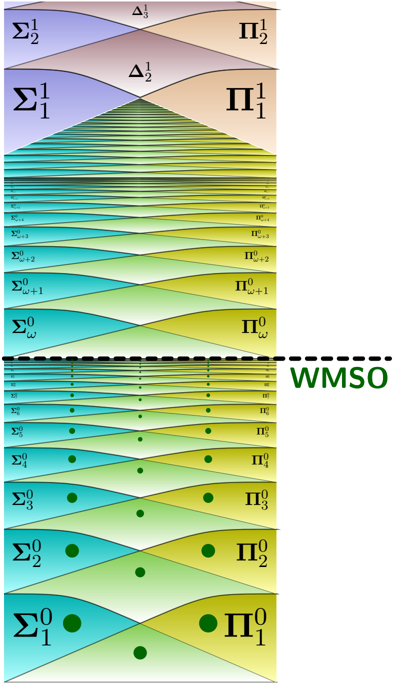
Upper bounds

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



Upper bounds

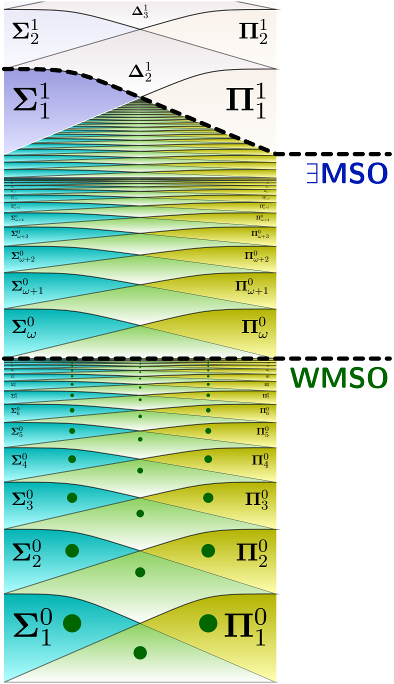
$$L \in \textbf{WMSO} \implies L \in \Sigma^0_n \quad \text{(for some } n\text{)}$$



Upper bounds

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

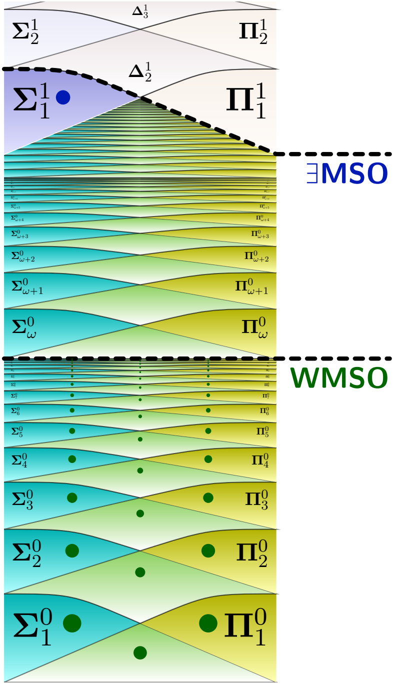
$$L \in \exists\textbf{MSO} \implies L \in \mathbf{\Sigma}^1_1$$



Upper bounds

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

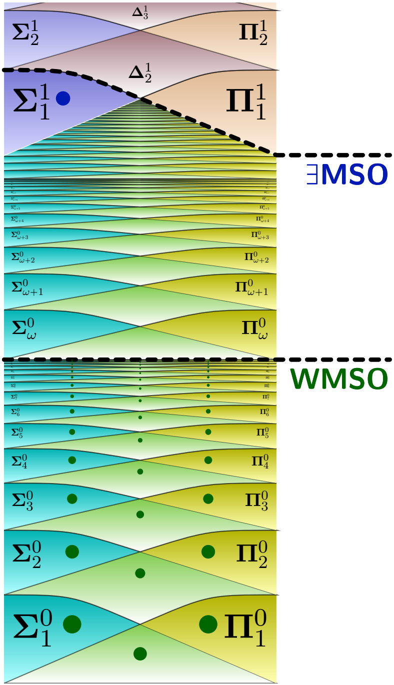
$$L \in \exists\text{MSO} \implies L \in \boldsymbol{\Sigma}^1_1$$



Upper bounds

$L \in$ WMSO $\implies L \in \Sigma^0_n$ (for some n)

$L \in \exists$MSO $\implies L \in \Sigma^1_1$

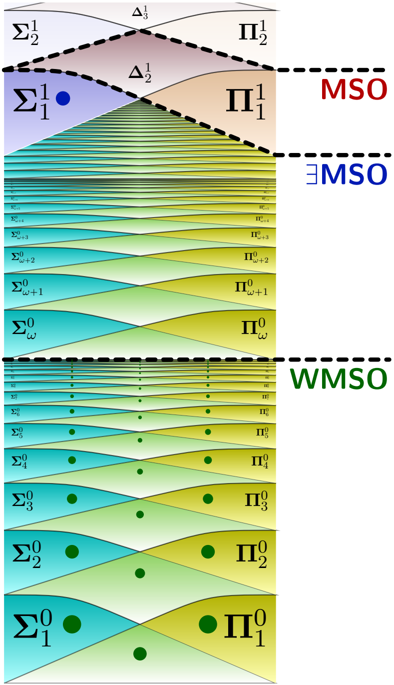


Upper bounds

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

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

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

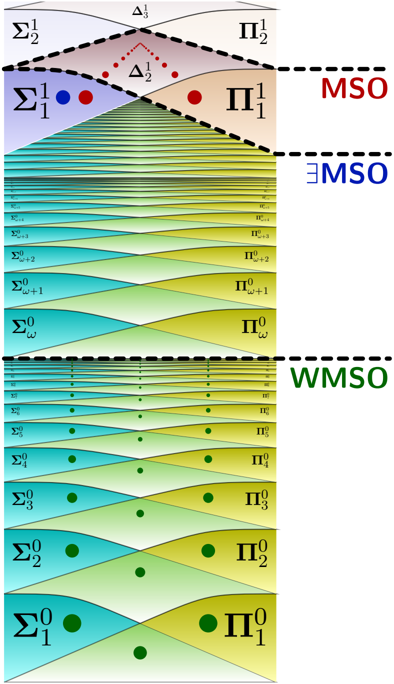


Upper bounds

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

$L \in$ **∃MSO** $\implies L \in \mathbf{\Sigma}^1_1$

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

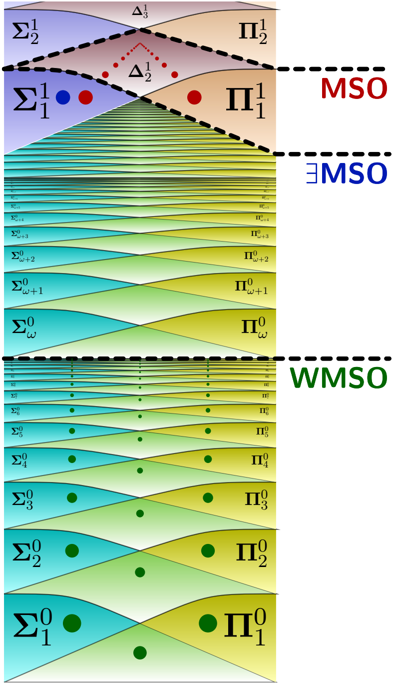


Upper bounds

$L \in$ WMSO $\implies L \in \Sigma^0_n$ (for some n)

$L \in$ ∃MSO $\implies L \in \Sigma^1_1$

$L \in$ MSO $\implies L \in \Delta^1_2$



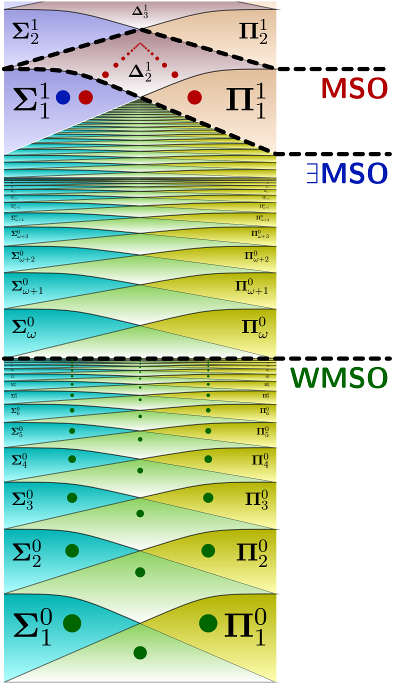
Upper bounds

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

$L \in$ ∃MSO $\implies$ $L \in \boldsymbol{\Sigma}^1_1$

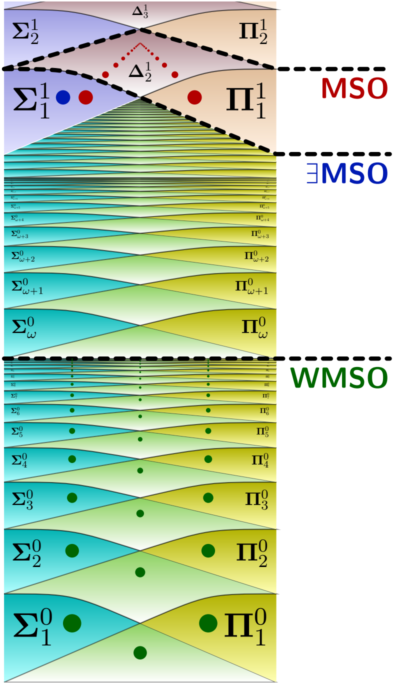
$L \in$ MSO $\implies$ $L \in \boldsymbol{\Delta}^1_2$

WMSO $\subsetneq$ ∃MSO $\subsetneq$ MSO

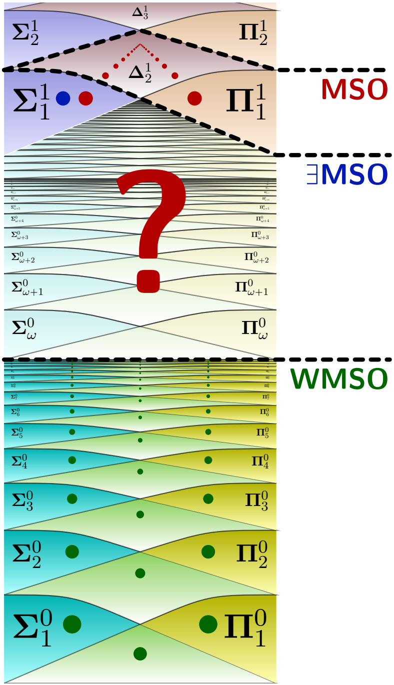


Gap property

Gap property



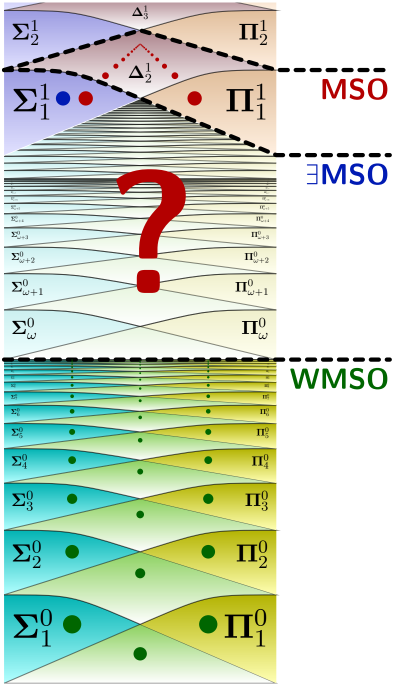
Gap property



Gap property

Conjecture (Skurczyński [1993])

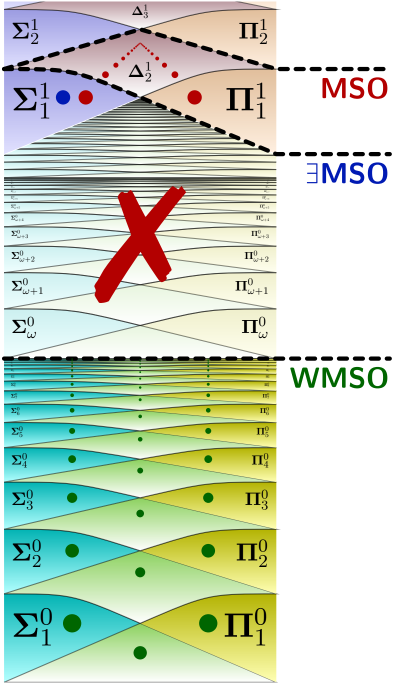
A Borel MSO-definable language is WMSO-definable.



Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language is WMSO-definable.



Gap property

Conjecture (Skurczyński [1993])

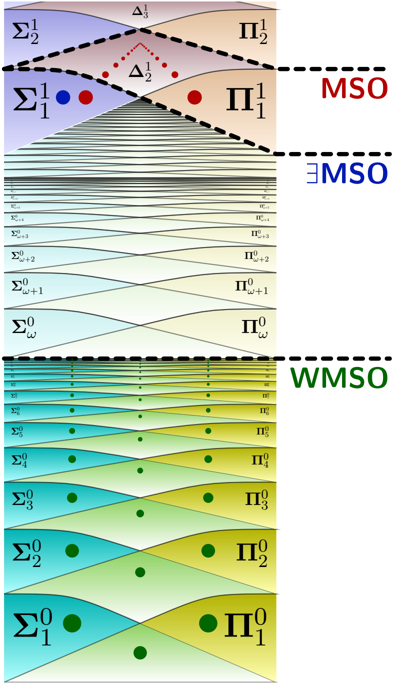
A Borel MSO-definable language is WMSO-definable.

Theorem (Walukiewicz, S. [2016])

An ∃MSO-definable language is either:

— WMSO-definable and Borel

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



Gap property

Conjecture (Skurczyński [1993])

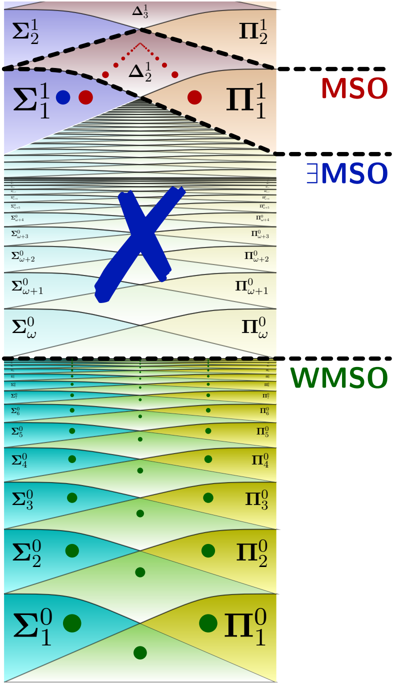
A Borel MSO-definable language is WMSO-definable.

Theorem (Walukiewicz, S. [2016])

An ∃MSO-definable language is either:

— WMSO-definable and Borel

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



Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language is WMSO-definable.

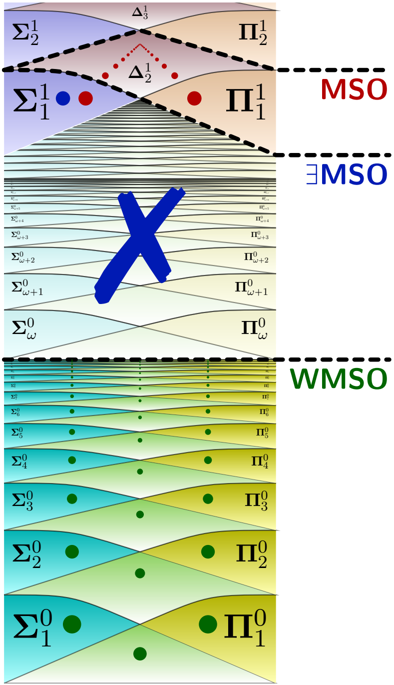
Theorem (Walukiewicz, S. [2016])

An ∃MSO-definable language is either:

— WMSO-definable and Borel

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

And the dychotomy is decidable.



Gap property

Conjecture (Skurczyński [1993])

A Borel MSO-definable language is WMSO-definable.

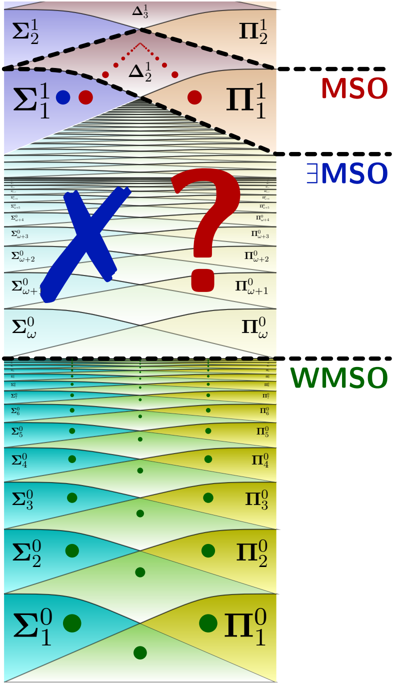
Theorem (Walukiewicz, S. [2016])

An ∃MSO-definable language is either:

— WMSO-definable and Borel

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

And the dychotomy is decidable.



Theorem (Walukiewicz, S. [2016])

$L \in$ ∃MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Theorem (Walukiewicz, S. [2016])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ_1^1-complete

Proof

Theorem (Walukiewicz, S. [2016])

$L \in$ **∃MSO** is effectively either:

— **WMSO**-definable

— Σ^1_1-complete

Proof

Input $\varphi \in$ **∃MSO** (or a Büchi automaton) for L

Theorem (Walukiewicz, S. [2016])

$L \in$ ∃MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Proof

Input $\varphi \in$ ∃MSO (or a Büchi automaton) for L

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

Theorem (Walukiewicz, S. [2016])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Proof

Input $\varphi \in \exists$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])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Proof

Input $\varphi \in \exists$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])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Proof

Input $\varphi \in \exists$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])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Proof

Input $\varphi \in \exists$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 $\rightsquigarrow$ $L \in$ **WMSO**

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

Theorem (Walukiewicz, S. [2016])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Proof

Input $\varphi \in \exists$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 $\rightsquigarrow$ $L \in$ **WMSO**

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

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

Theorem (Walukiewicz, S. [2016])

$L \in$ **∃MSO** is effectively either:

— **WMSO**-definable

— Σ^1_1-complete

Proof

Input $\varphi \in$ **∃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 $\rightsquigarrow L \in$ **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])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ_1^1-complete

Proof

Input $\varphi \in \exists$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 $\rightsquigarrow L \in$ **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])

$L \in$ ∃MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

Proof

Input $\varphi \in$ ∃MSO (or a Büchi automaton) for L

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

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

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

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

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

$\sigma_\exists$ — a winning strategy $\rightsquigarrow$ L is Σ^1_1-comp.

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

Summary

Summary

Theorem (CKLV [2013])

It is decidable for $L \in$ **∃MSO**

if $L \in$ **WMSO**.

Summary

Theorem (CKLV [2013])

It is decidable for $L \in \exists\text{MSO}$

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

Proof

Summary

Theorem (CKLV [2013])

It is decidable for $L \in \exists\textsc{mso}$ if $L \in \textsc{wmso}$.

Proof

Regular cost functions

Summary

Theorem (CKLV [2013])

It is decidable for $L \in \exists$MSO
if $L \in$ 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

— Σ^1_1-complete

Summary

Theorem (CKLV [2013])

It is decidable for $L \in \exists\textsc{mso}$
if $L \in \textsc{wmso}$.

Proof

Regular cost functions
+
Boundedness problem

Theorem (Walukiewicz, S. [2016])

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

— $\textsc{wmso}$-definable

— Σ^1_1-complete

(direct constructions)

Summary

Theorem (CKLV [2013])

It is decidable for $L \in \exists$MSO
if $L \in$ WMSO.

Proof

Regular cost functions
+
Boundedness problem

Theorem (Walukiewicz, S. [2016])

$L \in \exists$MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

(direct constructions)

(EXPTIME algorithm)

Summary

Theorem (CKLV [2013])

It is decidable for $L \in$ ∃MSO if $L \in$ WMSO.

Proof

Regular cost functions

+

Boundedness problem

Theorem (Walukiewicz, S. [2016])

$L \in$ ∃MSO is effectively either:

— WMSO-definable

— Σ^1_1-complete

(direct constructions)

($\mathrm{EXPTIME}$ algorithm)