Expressive power of Cost Logics over Infinite Words

Denis Kuperberg and Michael Vanden Boom

Séminaire Automates
24-02-2012

Introduction

- Regular cost functions: counting extension of regular languages

Introduction

- Regular cost functions: counting extension of regular languages
- Motivation: solving bound-related problems on regular languages (e.g. star-height)

Introduction

- Regular cost functions: counting extension of regular languages
- Motivation: solving bound-related problems on regular languages (e.g. star-height)
- Definable over finite or infinite structures, like words or trees

Introduction

- Regular cost functions: counting extension of regular languages
- Motivation: solving bound-related problems on regular languages (e.g. star-height)
- Definable over finite or infinite structures, like words or trees
- Definable via automata, logics, algebraic structures,...

Cost automata over infinite words

Nondeterministic finite-state automaton $\mathcal{A}$

$+$ finite set of counters

(initialized to 0, values range over $\mathbb{N}$)

$+$ counter operations on transitions

(increment $\mathtt{I}$, reset $\mathtt{R}$, check $\mathtt{C}$, no change ε)

Semantics

$[\![\mathcal{A}]\!] : \mathbb{A}^\omega \to \mathbb{N} \cup \{\infty\}$

Cost automata over infinite words

Nondeterministic finite-state automaton $\mathcal{A}$
+ finite set of counters
(initialized to 0, values range over $\mathbb{N}$)
+ counter operations on transitions
(increment `I`, reset `R`, check `C`, no change ε)

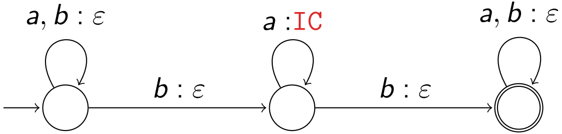
Semantics

$$val_B(\rho) := \text{max checked counter value during run } \rho$$

$$[\![\mathcal{A}]\!]_B(u) := \min\{val_B(\rho) : \rho \text{ is an accepting run of } \mathcal{A} \text{ on } u\}$$

Example

$[\![\mathcal{A}]\!]_B(u) =$ min length of block of a's surrounded by b's in u



Cost automata over infinite words

Nondeterministic finite-state automaton $\mathcal{A}$
$+$ finite set of counters
(initialized to 0, values range over $\mathbb{N}$)
$+$ counter operations on transitions
(increment $\mathtt{I}$, reset $\mathtt{R}$, check $\mathtt{C}$, no change ε)

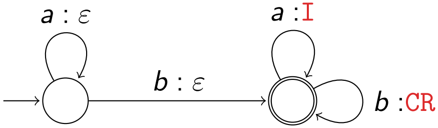
Semantics

$$val_S(\rho) := \text{min checked counter value during run } \rho$$

$$[\![\mathcal{A}]\!]_S(u) := \max\{val_S(\rho) : \rho \text{ is an accepting run of } \mathcal{A} \text{ on } u\}$$

Example

$[\![\mathcal{A}]\!]_S(u) =$ min length of block of a's surrounded by b's in u



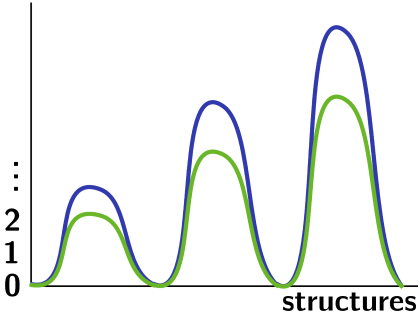
Boundedness relation

"$[\![\mathcal{A}]\!] = [\![\mathcal{B}]\!]$": undecidable [Krob '94]

Boundedness relation

"$[\![\mathcal{A}]\!] = [\![\mathcal{B}]\!]$": undecidable [Krob '94]

"$[\![\mathcal{A}]\!] \approx [\![\mathcal{B}]\!]$": decidable on words
[Colcombet '09, following Bojánczyk+Colcombet '06]
for all subsets U, $[\![\mathcal{A}]\!](U)$ bounded iff $[\![\mathcal{B}]\!](U)$ bounded

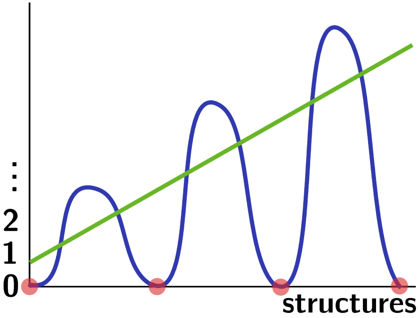


$$[\![\mathcal{A}]\!] \approx [\![\mathcal{B}]\!]$$

Boundedness relation

"$[\![\mathcal{A}]\!] = [\![\mathcal{B}]\!]$": undecidable [Krob '94]

"$[\![\mathcal{A}]\!] \approx [\![\mathcal{B}]\!]$": decidable on words
[Colcombet '09, following Bojánczyk+Colcombet '06]
for all subsets U, $[\![\mathcal{A}]\!](U)$ bounded iff $[\![\mathcal{B}]\!](U)$ bounded



$$[\![\mathcal{A}]\!] \not\approx [\![\mathcal{B}]\!]$$

Applications

Many problems for a regular language L can be reduced to deciding $\approx$ for some class of automata with counting features:

- **Finite power property** (finite words)
  [Simon '78, Hashiguchi '79]

  is there some n such that $(L + \epsilon)^n = L^*$?

- **Star-height problem** (finite words/trees)
  [Hashiguchi '88, Kirsten '05, Colcombet+Löding '08]

  given n, is there a regular expression for L
  with at most n nestings of Kleene star?

- **Parity-index problem** (infinite trees)
  [reduction in Colcombet+Löding '08, decidability open]

  given $i < j$, is there a parity automaton for L
  which uses only priorities $\{i, i+1, \ldots, j\}$?

Applications

Many problems for a regular language L can be reduced to deciding $\approx$ for some class of automata with counting features:

- **Finite power property** (finite words)
  [Simon '78, Hashiguchi '79]

  distance

  is there some n such that $(L+\epsilon)^n = L^*$?

- **Star-height problem** (finite words/trees)
  [Hashiguchi '88, Kirsten '05, Colcombet+Löding '08]

  nested distance-desert

  given n, is there a regular expression for L with at most n nestings of Kleene star?

- **Parity-index problem** (infinite trees)
  [reduction in Colcombet+Löding '08, decidability open]

  cost-parity

  given $i < j$, is there a parity automaton for L which uses only priorities $\{i, i+1, \dots, j\}$?

Languages as cost functions

- A standard automaton $\mathcal{A}$ computing a language L can be viewed as a B- or S-automaton without any counters. Then $[\![\mathcal{A}]\!]_B = \chi_L$ and $[\![\mathcal{A}]\!]_S = \chi_{\overline{L}}$, with

$$\chi_L(u) = \begin{cases} 0 & \text{if } u \in L \\ \infty & \text{if } u \notin L \end{cases}$$

Languages as cost functions

- A standard automaton $\mathcal{A}$ computing a language L can be viewed as a B- or S-automaton without any counters. Then $[\![\mathcal{A}]\!]_B = \chi_L$ and $[\![\mathcal{A}]\!]_S = \chi_{\overline{L}}$, with

$$\chi_L(u) = \begin{cases} 0 & \text{if } u \in L \\ \infty & \text{if } u \notin L \end{cases}$$

- Switching between B and S semantics corresponds to a complementation.

Languages as cost functions

- A standard automaton $\mathcal{A}$ computing a language L can be viewed as a B- or S-automaton without any counters. Then $[\![\mathcal{A}]\!]_B = \chi_L$ and $[\![\mathcal{A}]\!]_S = \chi_{\overline{L}}$, with

$$\chi_L(u) = \begin{cases} 0 & \text{if } u \in L \\ \infty & \text{if } u \notin L \end{cases}$$

- Switching between B and S semantics corresponds to a complementation.
- If L and L' are languages, $\chi_L \approx \chi_{L'}$ iff $L = L'$, so cost function theory, even up to $\approx$, strictly extends language theory.

Languages as cost functions

- A standard automaton $\mathcal{A}$ computing a language L can be viewed as a B- or S-automaton without any counters. Then $[\![\mathcal{A}]\!]_B = \chi_L$ and $[\![\mathcal{A}]\!]_S = \chi_{\overline{L}}$, with

$$\chi_L(u) = \begin{cases} 0 & \text{if } u \in L \\ \infty & \text{if } u \notin L \end{cases}$$

- Switching between B and S semantics corresponds to a complementation.
- If L and L' are languages, $\chi_L \approx \chi_{L'}$ iff $L = L'$, so cost function theory, even up to $\approx$, strictly extends language theory.
- **Aim:** Extend classic theorems from languages to cost functions

Cost functions on infinite words

- In the following, input structures = $\mathbb{A}$-labelled infinite words.

Cost functions on infinite words

- In the following, input structures $=$ $\mathbb{A}$-labelled infinite words.
- Dual B- and S- semantics as before, defining functions: $\mathbb{A}^\omega \to \mathbb{N} \cup \{\infty\}$.

Cost functions on infinite words

- In the following, input structures $=$ $\mathbb{A}$-labelled infinite words.
- Dual B- and S- semantics as before, defining functions: $\mathbb{A}^\omega \to \mathbb{N} \cup \{\infty\}$.
- We aim at extending classical theorems on languages to the setting of cost functions.

Logics on infinite words

- LTL on $\mathbb{A}$ describes regular languages:
  $\varphi := a \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \mathbf{R} \varphi \mid \varphi \mathbf{U} \varphi$
  where the negations have been pushed to the leaves, and the **U** corresponds to "Next Until".

  $$\varphi U \psi: \qquad \begin{array}{ccccccccccc} \varphi & \varphi & \varphi & \varphi & \varphi & \varphi & \varphi & \psi & & & \\ a_0 & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} \end{array}$$

  We can define **X** (Next), **G** (Always) and **F** (Eventually) in terms of these operators.
- First-Order Logic (FO):

  $$\varphi := a(x) \mid x = y \mid x < y \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x. \varphi \mid \forall x. \varphi$$

- (Weak) MSO: FO with quantification over (finite) sets, set variables noted X, Y.

Cost LTL

- CLTL on $\mathbb{A}$ describes regular cost functions:
  $\varphi := a \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \mathbf{R} \varphi \mid \varphi \mathbf{U} \varphi \mid \varphi \mathbf{U}^{\leq N} \varphi$

Cost LTL

- CLTL on $\mathbb{A}$ describes regular cost functions:
  $\varphi := a \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \mathbf{R} \varphi \mid \varphi \mathbf{U} \varphi \mid \varphi \mathbf{U}^{\leq N} \varphi$
- $\varphi \mathbf{U}^{\leq N} \psi$ means that ψ is true somewhere in the future, and φ is false at most N times until then.

$$\varphi \mathbf{U}^{\leq N} \psi: \qquad \begin{array}{ccccccccccc} \varphi & \times & \varphi & \varphi & \times & \varphi & \varphi & \psi & & & \\ a_0 & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} \end{array}$$

Cost LTL

- CLTL on $\mathbb{A}$ describes regular cost functions:
  $\varphi := a \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \mathbf{R} \varphi \mid \varphi \mathbf{U} \varphi \mid \varphi \mathbf{U}^{\leq N} \varphi$
- $\varphi \mathbf{U}^{\leq N} \psi$ means that ψ is true somewhere in the future, and φ is false at most N times until then.

$$\varphi \mathbf{U}^{\leq N} \psi: \qquad \begin{array}{ccccccccccc} \varphi & \times & \varphi & \varphi & \times & \varphi & \varphi & \psi & & & \\ a_0 & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} \end{array}$$

- The "error value" variable N is unique, and is shared by all occurrences of $\mathbf{U}^{\leq N}$ operator.

Cost LTL

- CLTL on $\mathbb{A}$ describes regular cost functions:
  $\varphi := a \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \mathbf{R} \varphi \mid \varphi \mathbf{U} \varphi \mid \varphi \mathbf{U}^{\leq N} \varphi$
- $\varphi \mathbf{U}^{\leq N} \psi$ means that ψ is true somewhere in the future, and φ is false at most N times until then.

$$\varphi \mathbf{U}^{\leq N} \psi: \qquad \begin{array}{ccccccccccc} \varphi & \times & \varphi & \varphi & \times & \varphi & \varphi & \psi & & & \\ a_0 & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} \end{array}$$

- The "error value" variable N is unique, and is shared by all occurrences of $\mathbf{U}^{\leq N}$ operator.
- $\mathbf{G}^{\leq N}$ and $\mathbf{R}^{\leq N}$ can be defined in terms of the previous operators.

CFO and CMSO

- CFO on $\mathbb{A}$ describes regular cost functions:

  $$\varphi := a(x) \mid x = y \mid x < y \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x.\varphi \mid \forall x.\varphi \mid \forall^{\leq N} x.\varphi$$

- As before, N is a unique free variable and counts the number of mistakes.
- (Weak) CMSO extends CFO with quantification over (finite) sets.

Semantics of Cost Logics

From formula to cost function:

$[\![\varphi]\!]$ is the cost function associated to φ, defined by

$$[\![\varphi]\!](u) = \inf\{N \in \mathbb{N}, \varphi \text{ is true on } u \text{ with N as error value}\}$$

Example

For all $u \in \{a, b\}^\omega$, we have

- $[\![b\mathbf{U}^{\leq N}(\mathbf{G}b)]\!](u) = [\![\forall^{\leq N} x.b(x)]\!](u) = |u|_a$.
- $[\![\mathbf{G}(\perp\mathbf{U}^{\leq N} b)]\!](u) = [\![\forall X, \mathrm{block}_a(X) \Rightarrow (\forall^{\leq N} x, x \notin X)]\!](u) = \mathrm{maxblock}_a(u)$

Alternating B-automata

- **Alternating B-automaton:** Game between Eve and Adam, with counter actions on transitions. Eve must satisfy acceptance condition AND low counter value.

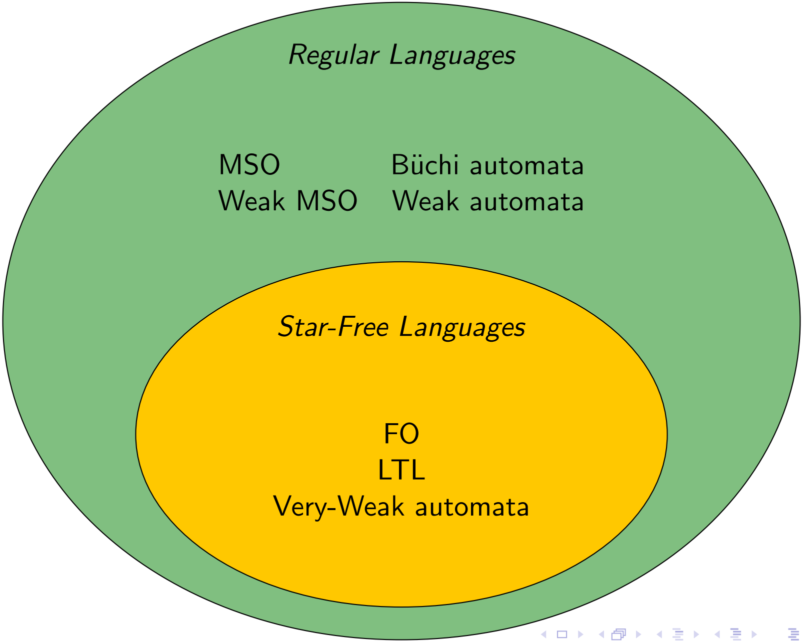
Alternating B-automata

- **Alternating B-automaton:** Game between Eve and Adam, with counter actions on transitions. Eve must satisfy acceptance condition AND low counter value.
- **Weak B-automaton:** Büchi condition, no cycle with both accepting and rejecting states.

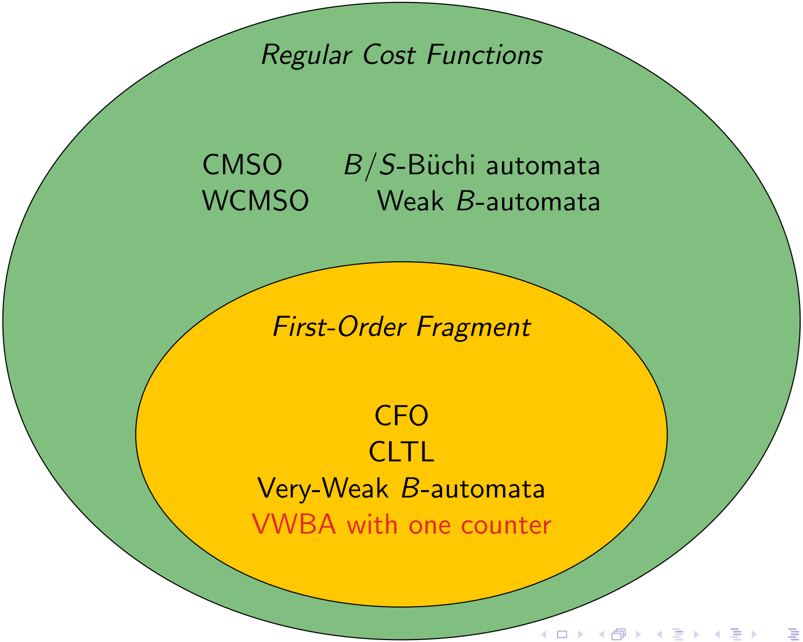
Alternating B-automata

- **Alternating B-automaton:** Game between Eve and Adam, with counter actions on transitions. Eve must satisfy acceptance condition AND low counter value.
- **Weak B-automaton:** Büchi condition, no cycle with both accepting and rejecting states.
- **Very-weak B-automaton:** Büchi condition, no non-trivial cycle.

Classical picture



Cost Functions

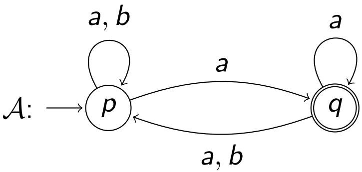


Proof ideas for WCMSO to CMSO

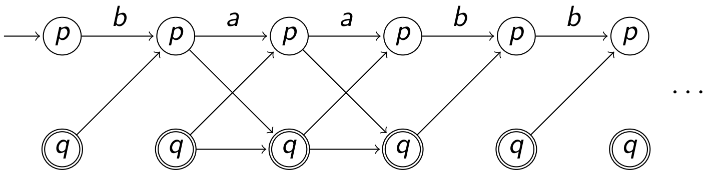
- By Colcombet, CMSO $\Leftrightarrow$ nondeterministic B/S-Büchi automata.
- By [Vanden Boom 11], WCMSO $\Leftrightarrow$ weak alternating B-automata.

We just need to show a translation nondeterministic B-Büchi automata $\rightarrow$ weak alternating B-automata.

Classical Proof from [Kupferman+Vardi '01]

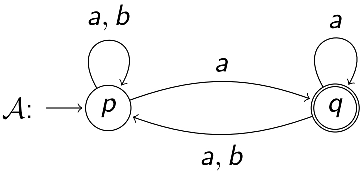


Fix a word u, and analyze the run-DAG of the Büchi-automaton on u (here for $u = baab^\omega$):

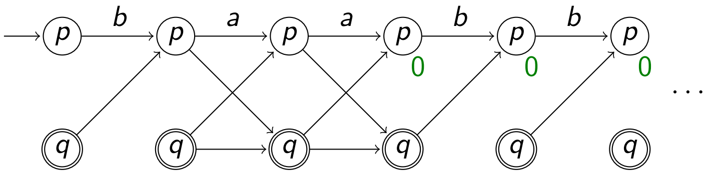


Ranks : No more Büchi or finite path on the remaining DAG. Initial node gets a rank $\Rightarrow$ u is rejected.

Classical Proof from [Kupferman+Vardi '01]

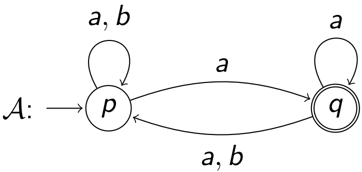


Fix a word u, and analyze the run-DAG of the Büchi-automaton on u (here for $u = baab^\omega$):

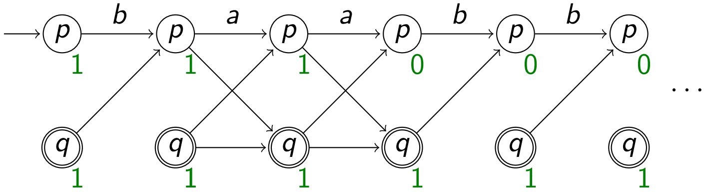


Ranks : No more Büchi or finite path on the remaining DAG.
Initial node gets a rank $\Rightarrow$ u is rejected.

Classical Proof from [Kupferman+Vardi '01]

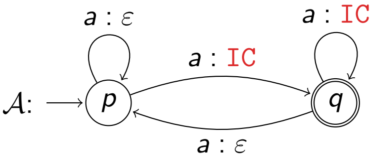


Fix a word u, and analyze the run-DAG of the Büchi-automaton on u (here for $u = baab^{\omega}$):

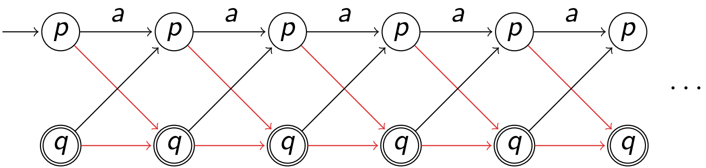


Ranks : No more Büchi or finite path on the remaining DAG.
Initial node gets a rank $\Rightarrow$ u is rejected.

Extending to cost functions



Run-DAG for $u = a^\omega$:



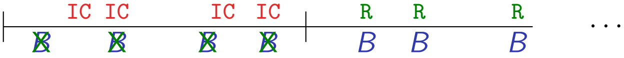
Problem to assign ranks : how to prove that this run has value ∞ ?

Solution :

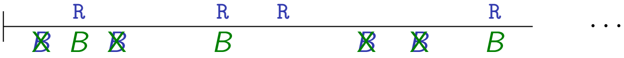
Normal form for nondeterministic B-Büchi automata : must do a reset on every counter after each Büchi state.

The modified automaton guesses whether there is

- a finite number of increments $\Rightarrow$ add resets to Büchi states:



- infinitely many resets $\Rightarrow$ delay Büchi states after the next reset (equivalent up to $\approx$):



On these Büchi automata in normal form, we can define ranks in a sound way, for each value n.

Description of the weak alternating automaton

The weak B-automaton W describes a game between two players:

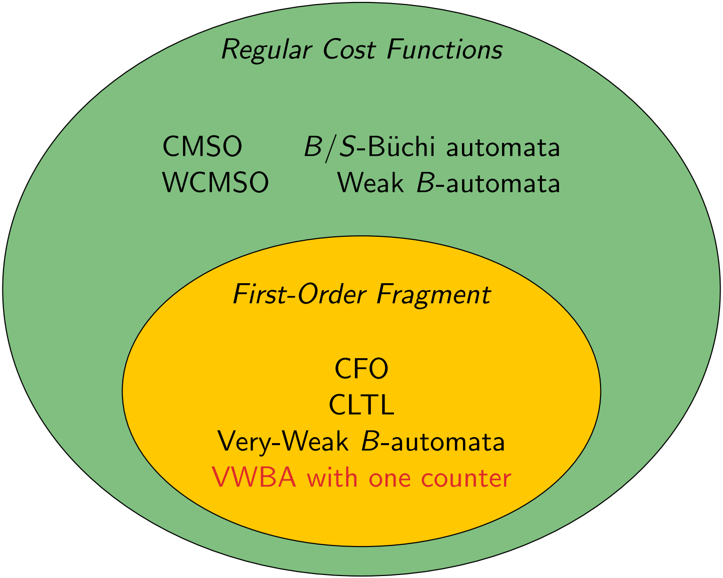
- Eve wants to prove that $\mathcal{A}$ accepts with low value
- Adam wants to prove that this is not the case

It allows Eve to play a run of $\mathcal{A}$, and Adam to guess ranks. It is designed in such a way that

- playing a n-run (if exists) is a strategy of value $\leq n$ for Eve.
- playing the n-ranks (if exists) is a strategy of value $> n$ for Adam.

From this we get $[\![W]\!] = [\![\mathcal{A}]\!]$.

Summary



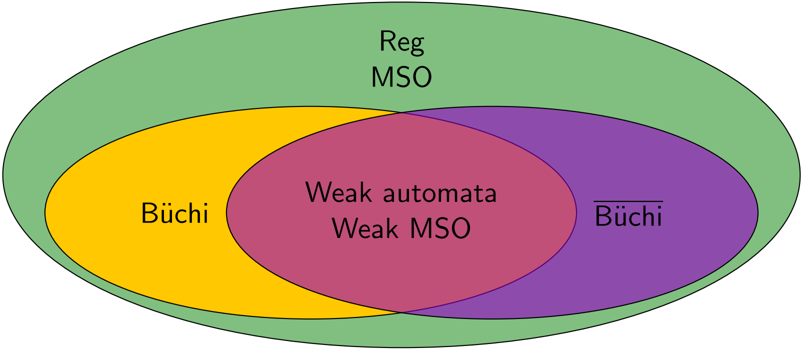
What are the limits of this correspondence ?

On Infinite trees

Theorem (Rabin 1970, Kupferman + Vardi 1999)

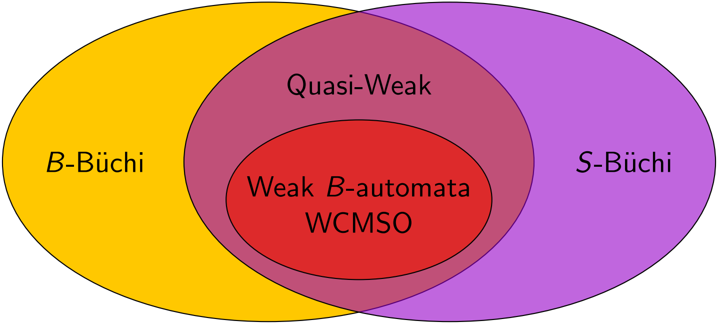
A language L of infinite trees is recognizable by
an alternating weak automaton iff there are nondeterministic Büchi automata $\mathcal{U}$ and $\mathcal{U}'$ such that

$$L = L(\mathcal{U}) = \overline{L(\mathcal{U}')}.$$

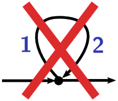


Extension to Cost Functions

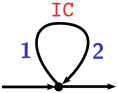
Complementation becomes switching between B- and S-semantic:



Weak:



Quasi-Weak:



Inclusions are strict [K.+Vanden Boom '11]

Thank you !