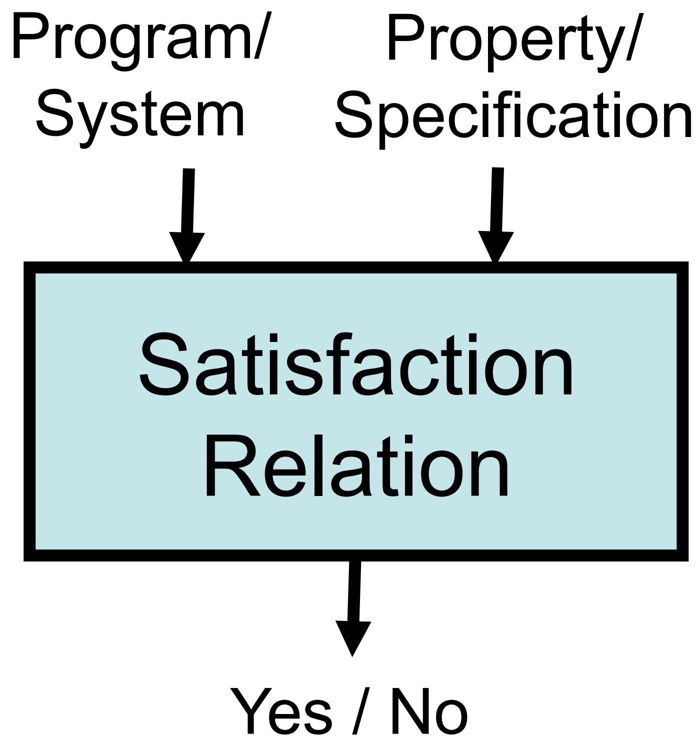
Quantitative Languages

Laurent Doyen
LSV, ENS Cachan & CNRS

joint work with Krishnendu Chatterjee and
Tom Henzinger (IST Austria)

(CSL'08, LICS'09)

Model-Checking



-perhaps a proof
-perhaps some counterexamples

Model-Checking

Finite
automaton

Program/
System

Property/
Specification

Formula

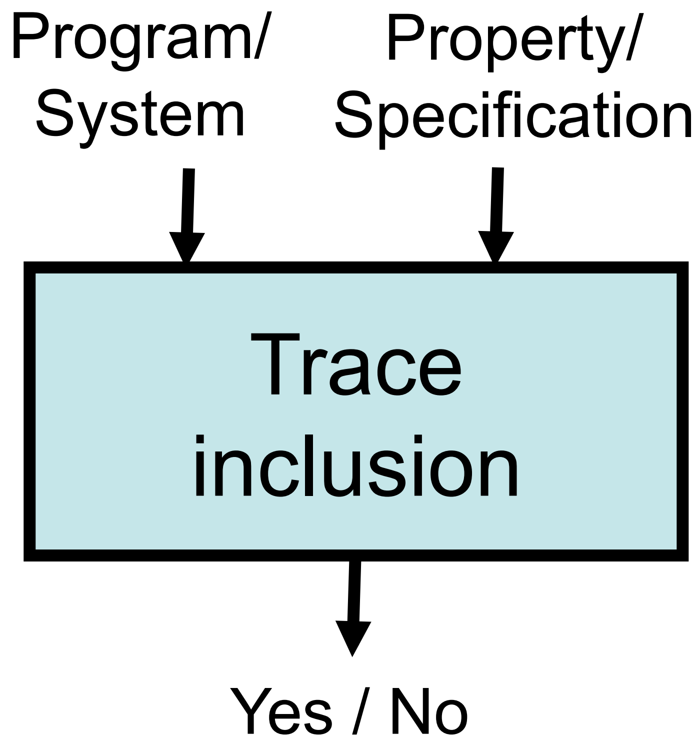
Every request is
followed by a grant

Trace
inclusion

Yes / No

Model-Checking

Finite
automaton



Formula

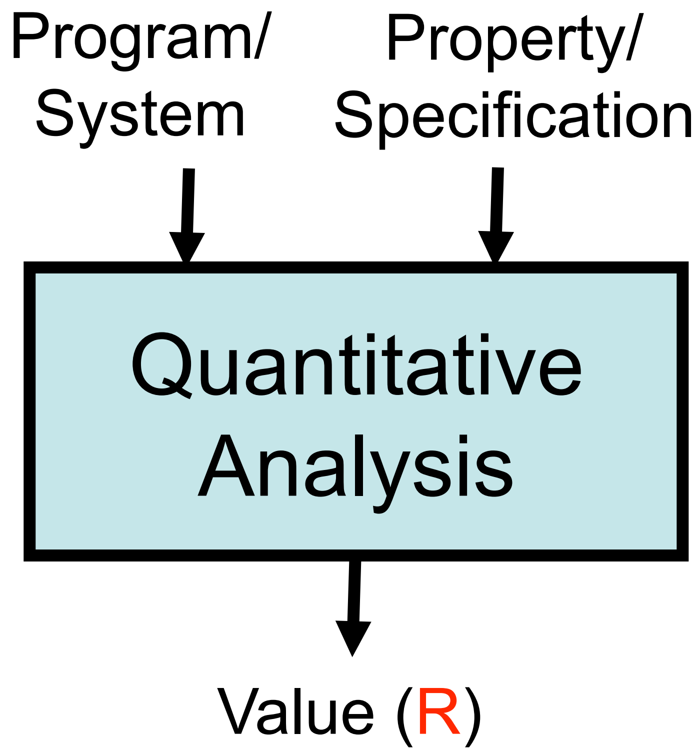
Every request is
followed by a grant

Model-checking is boolean

- a trace is either good or bad

Quantitative Analysis

Finite automaton



Formula

Every request is followed by a grant

-Measure of “fit” between system and spec

-e.g. average number of requests immediately granted

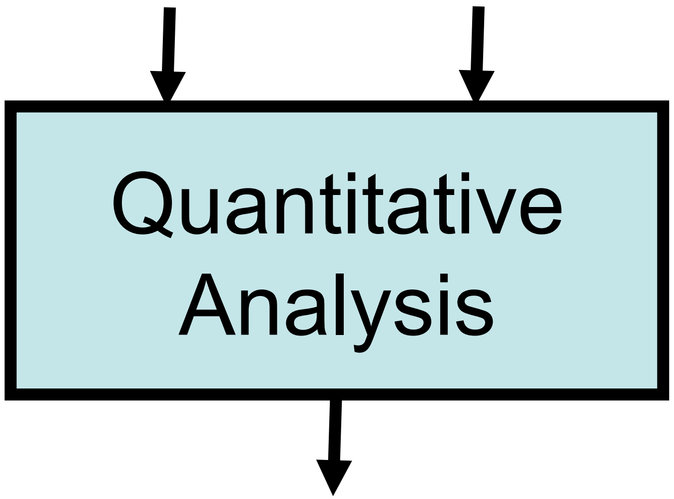
Quantitative Analysis

Finite automaton

Program/
System #1

Program/
System #2

Every request is followed by a grant



Distance (R)

- Comparing two implementations

e.g. cost or quality measure

Quantitative Model-checking

Is there a Quantitative Framework with

- an appealing mathematical formulation,
- useful expressive power, and
- good algorithmic properties ?

(Like the boolean theory of ω-regularity.)

Note: “Quantitative” is more than “timed” and “probabilistic”

Quantitative languages

A language is a boolean function:

$$\boxed{L : \Sigma^\omega \to \{0, 1\}}$$

Quantitative languages

A language is a boolean function:

$$\boxed{L : \Sigma^\omega \to \{0, 1\}}$$

A quantitative language is a function:

$$\boxed{L : \Sigma^\omega \to \mathbb{R}}$$

L(w) can be interpreted as:

- the amount of some resource needed by the system to produce w (power, energy, time consumption),
- a reliability measure (the average number of “faults” in w).

Quantitative languages

Quantitative language inclusion

Is $L_A(w) \leq L_B(w)$ for all words w ?

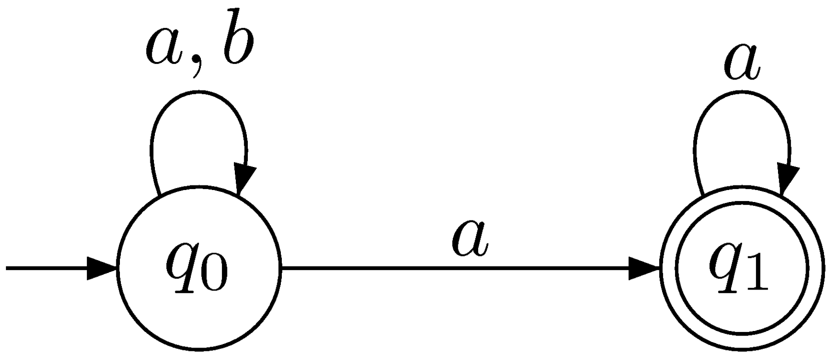
$L_A(w)$	$L_B(w)$
peak resource consumption	resource bound
Long-run average response time	Average response-time requirement

Outline

- Motivation
- Weighted automata
- Decision problems
- Expressive power
- Closure properties

Automata

Boolean languages are generated by finite automata.

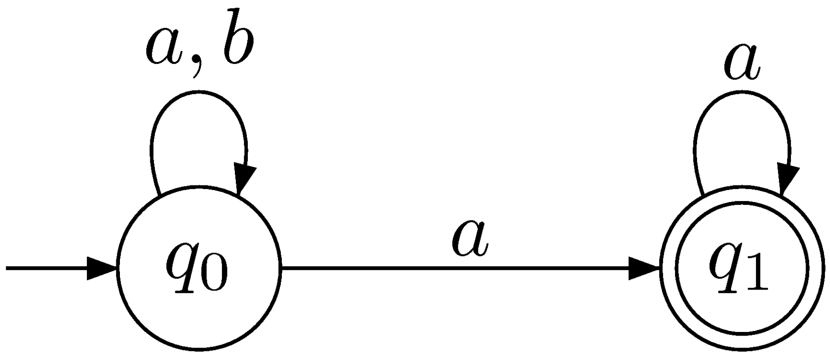


$\mathsf{L} = (a + b)^* \cdot a^\omega$

Nondeterministic Büchi automaton

Automata

Boolean languages are generated by finite automata.



$$L = (a+b)^* \cdot a^\omega$$

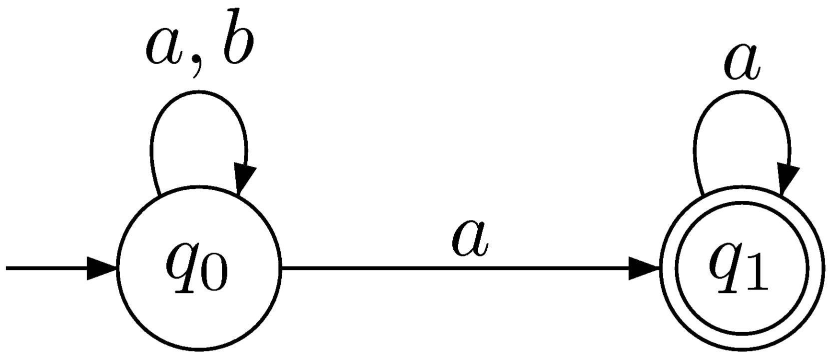
Nondeterministic Büchi automaton

Value of a run r: Val(r)= 1 if an accepting state occurs ∞-ly often in r

0 otherwise

Automata

Boolean languages are generated by finite automata.



$L = (a+b)^* \cdot a^\omega$

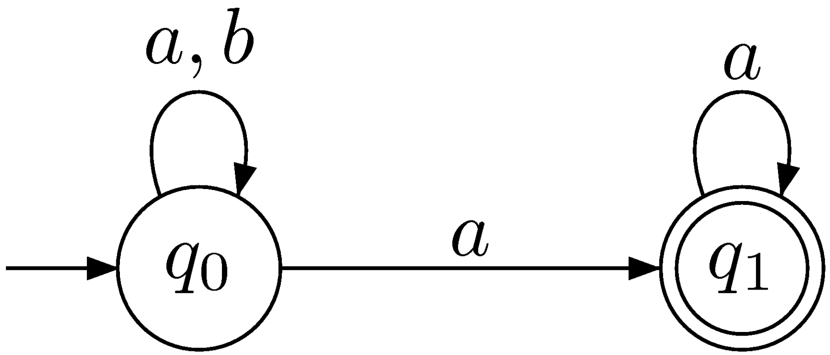
Nondeterministic Büchi automaton

Value of a run r: Val(r)= 1 if an accepting state occurs ∞-ly often in r

Value of a word w: max of {values of the runs r over w}

Automata

Boolean languages are generated by finite automata.



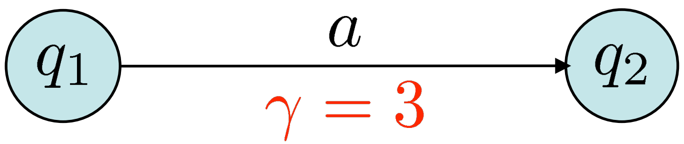
$\mathsf{L} = (a+b)^* \cdot a^\omega$

Nondeterministic Büchi automaton

$L_A(w)$ = max of {Val(r) | r is a run of A over w}

Weighted automata

Quantitative languages are generated by weighted automata.



Weight function $\gamma : Q \times \Sigma \times Q \to \mathbb{Q}$

Weighted automata

Quantitative languages are generated by weighted automata.

$$q_1 \xrightarrow[\gamma = 3]{a} q_2$$

Weight function $\gamma : Q \times \Sigma \times Q \to \mathbb{Q}$

Value of a word w: max of {values of the runs r over w}

Value of a run r: Val(r)

where $\mathrm{Val} : \mathbb{Q}^\omega \to \mathbb{R}$ is a value function

Some value functions

For $v = v_0 v_1 \ldots$ ($v_i \in \mathbb{Q}$), let

($v_i \in \{0,1\}$)

- $\mathsf{Sup}(v) = \sup\{v_n \mid n \geq 0\}$; (reachability)
- $\mathsf{LimSup}(v) = \limsup_{n\to\infty} v_n = \lim_{n\to\infty} \sup\{v_i \mid i \geq n\}$; (Büchi)
- $\mathsf{LimInf}(v) = \liminf_{n\to\infty} v_n = \lim_{n\to\infty} \inf\{v_i \mid i \geq n\}$; (coBüchi)

Some value functions

For $v = v_0 v_1 \dots$ $(v_i \in \mathbb{Q})$, let

($v_i \in \{0,1\}$)

- $\mathsf{Sup}(v) = \sup\{v_n \mid n \geq 0\}$; (reachability)
- $\mathsf{LimSup}(v) = \limsup\limits_{n\to\infty} \; v_n = \lim\limits_{n\to\infty} \sup\{v_i \mid i \geq n\}$; (Büchi)
- $\mathsf{LimInf}(v) = \liminf\limits_{n\to\infty} \; v_n = \lim\limits_{n\to\infty} \inf\{v_i \mid i \geq n\}$; (coBüchi)
- $\mathsf{LimAvg}(v) = \liminf\limits_{n\to\infty} \; \frac{1}{n} \cdot \sum_{i=0}^{n-1} v_i$;
- given $0 < \lambda < 1$, $\mathsf{Disc}_\lambda(v) = \sum_{i=0}^{\infty} \lambda^i \cdot v_i$.

Outline

- Motivation
- Weighted automata
- Decision problems
- Expressive power
- Closure properties

Emptiness

Given $\nu \in \mathbb{Q}$, is $L_A(w) \geq \nu$ for some word w ?

- solved by finding the maximal value of an infinite path in the graph of A,
- memoryless strategies exist in the corresponding quantitative 1-player games,
- decidable in polynomial time for Sup, LimSup, LimInf, LimAvg and Disc_λ.

Language Inclusion

Is $L_A(w) \leq L_B(w)$ for all words w ?

- PSPACE-complete for Sup, LimSup and LimInf.

- Solvable in polynomial-time when B is deterministic for LimAvg and Disc_λ,

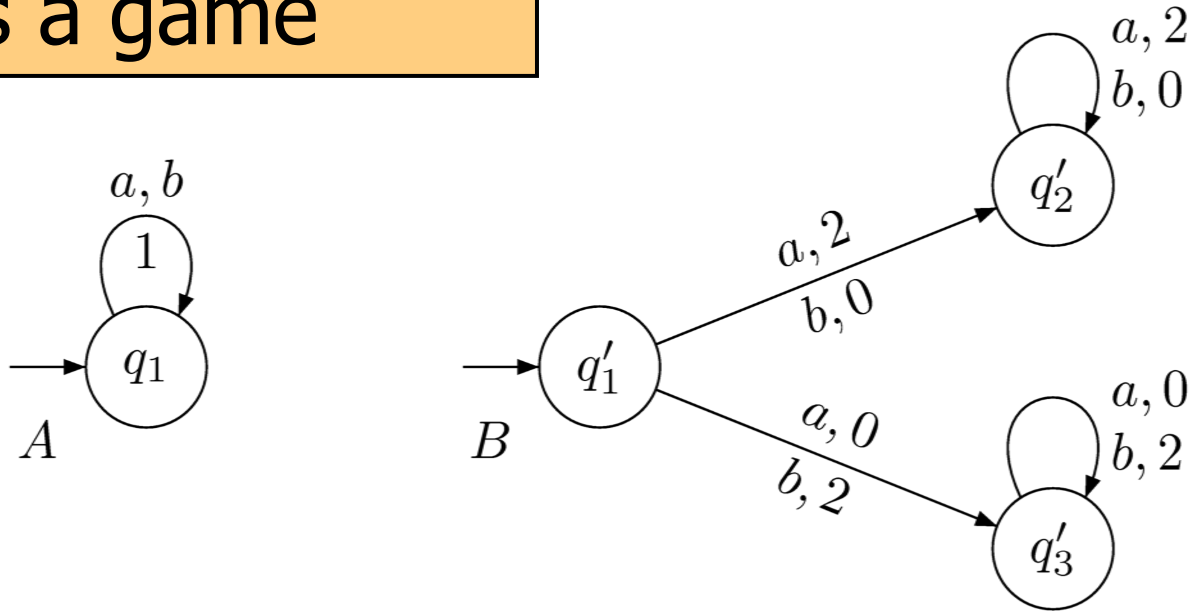
- open question for nondeterministic automata.

Language Inclusion

	Quant. L. inclusion	
Sup	PSpace	
LimSup	PSpace	
LimInf	PSpace	
LimAvg	?	
Disc_λ	?	

Language-inclusion game

Language inclusion as a game

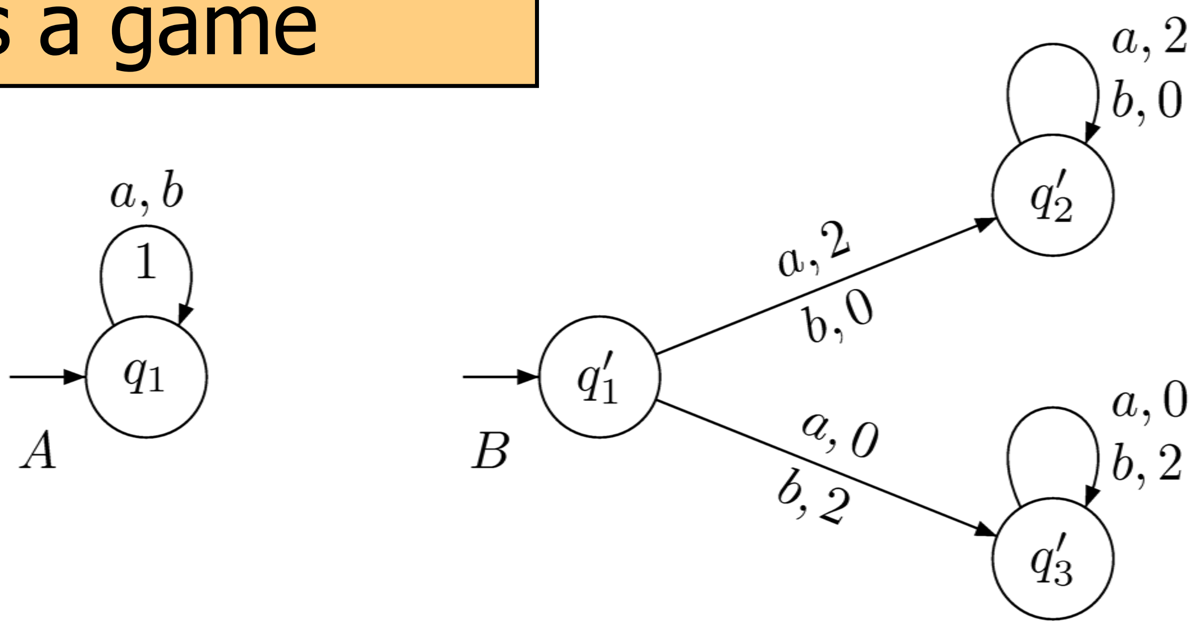


Discounted-sum automata, λ=3/4

Is $L_A(w) \leq L_B(w)$ for all words w ?

Language-inclusion game

Language inclusion as a game



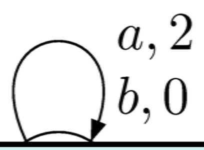
Discounted-sum automata, $\lambda=3/4$

Is $L_A(w) \leq L_B(w)$ for all words w ?

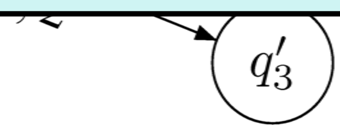
Yes !

Language-inclusion game

Language inclusion as a game



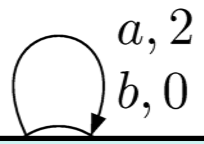
- turn-based, two players;
- Challenger wins iff $\exists w : L_A(w) > L_B(w)$.



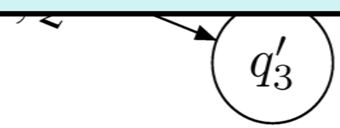
Is $L_A(w) \leq L_B(w)$ for all words w ?

Language-inclusion game

Language inclusion
as a game



- turn-based, two players;
- Challenger wins iff $\exists w : L_A(w) > L_B(w)$.

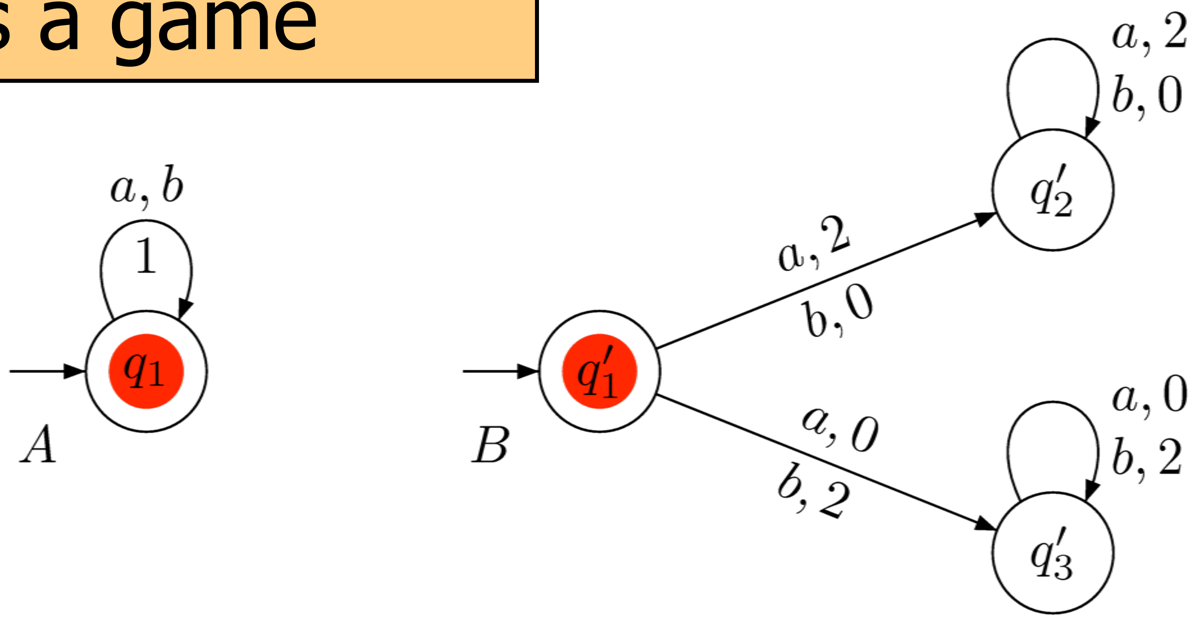


winning strategy

Is $L_A(w) \leq L_B(w)$ for all words w ?

Language-inclusion game

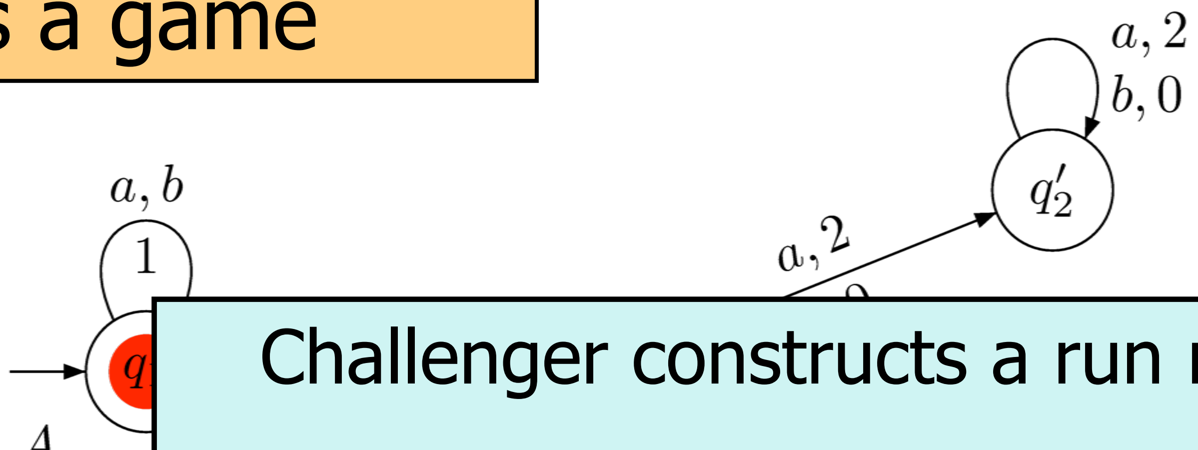
Language inclusion as a game



Tokens on the initial states

Language-inclusion game

Language inclusion
as a game



Challenger constructs a run r_1 of A,

Simulator constructs a run r_2 of B.

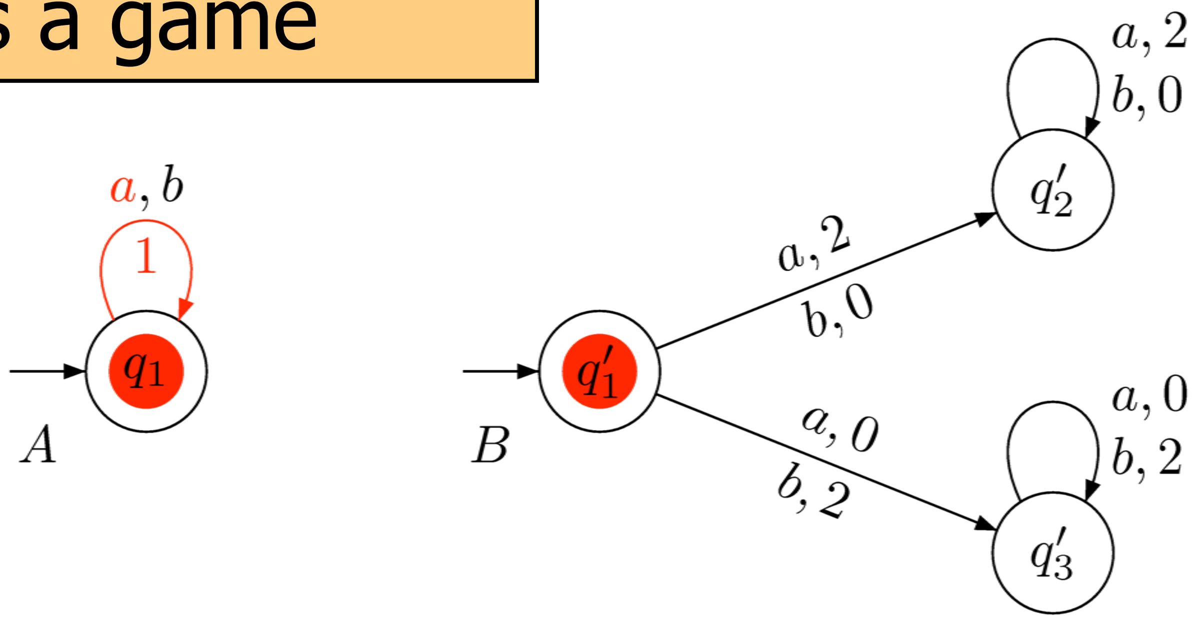
Challenger wins if $\mathrm{Val}(r_1) > \mathrm{Val}(r_2)$.

Challenger: q_1

Simulator: q_1'

Language-inclusion game

Language inclusion as a game

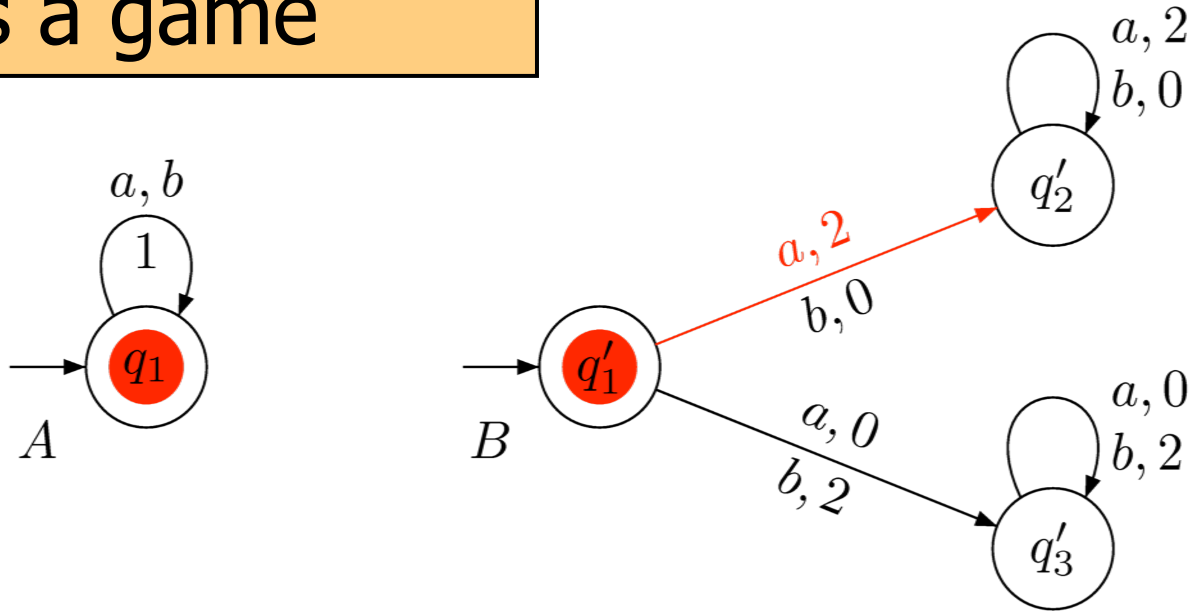


Challenger: $q_1 \xrightarrow[1]{a} q_1$

Simulator: q_1'

Language-inclusion game

Language inclusion as a game

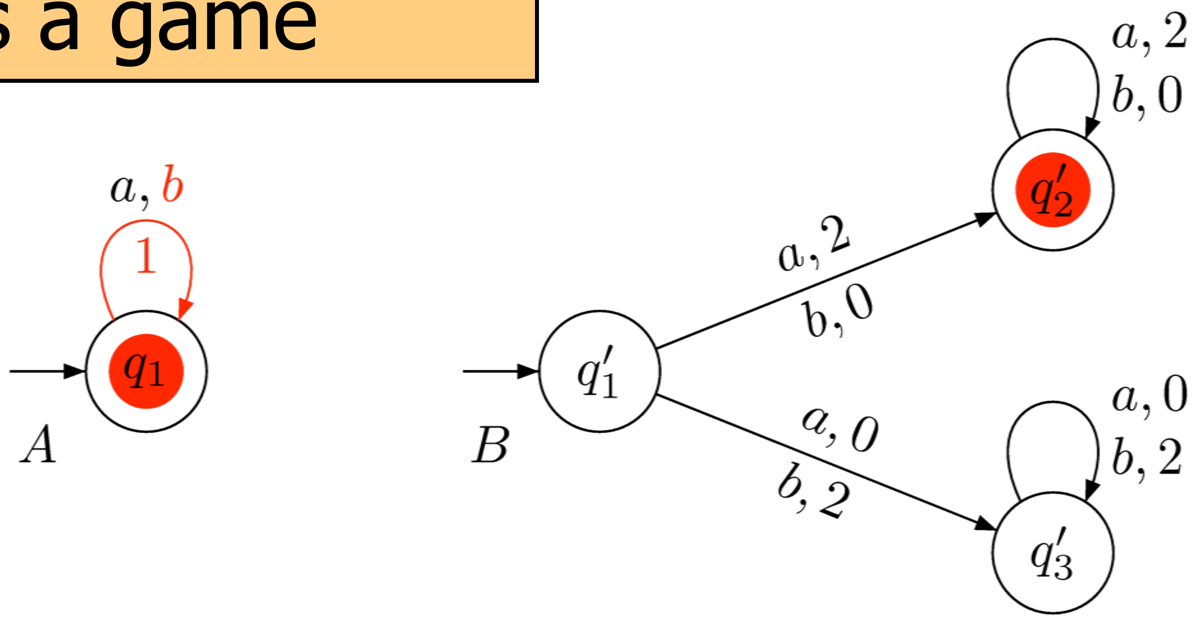


Challenger: $q_1 \xrightarrow[1]{a} q_1$

Simulator: $q'_1 \xrightarrow[2]{a} q'_2$

Language-inclusion game

Language inclusion as a game

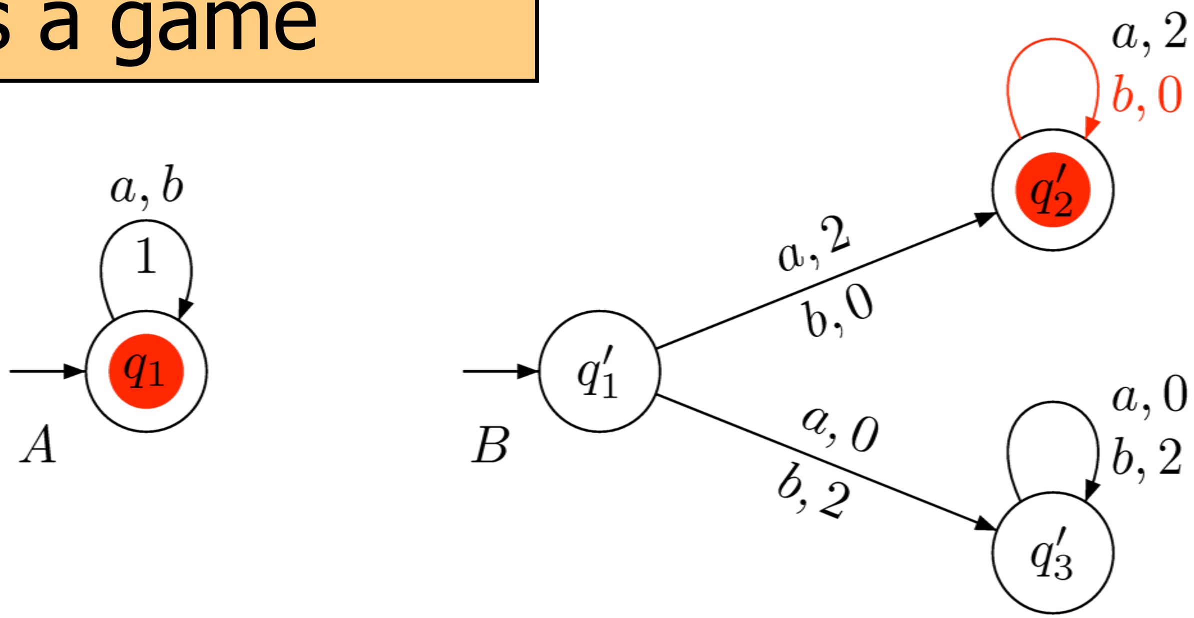


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1$

Simulator: $q'_1 \xrightarrow[2]{a} q'_2$

Language-inclusion game

Language inclusion as a game

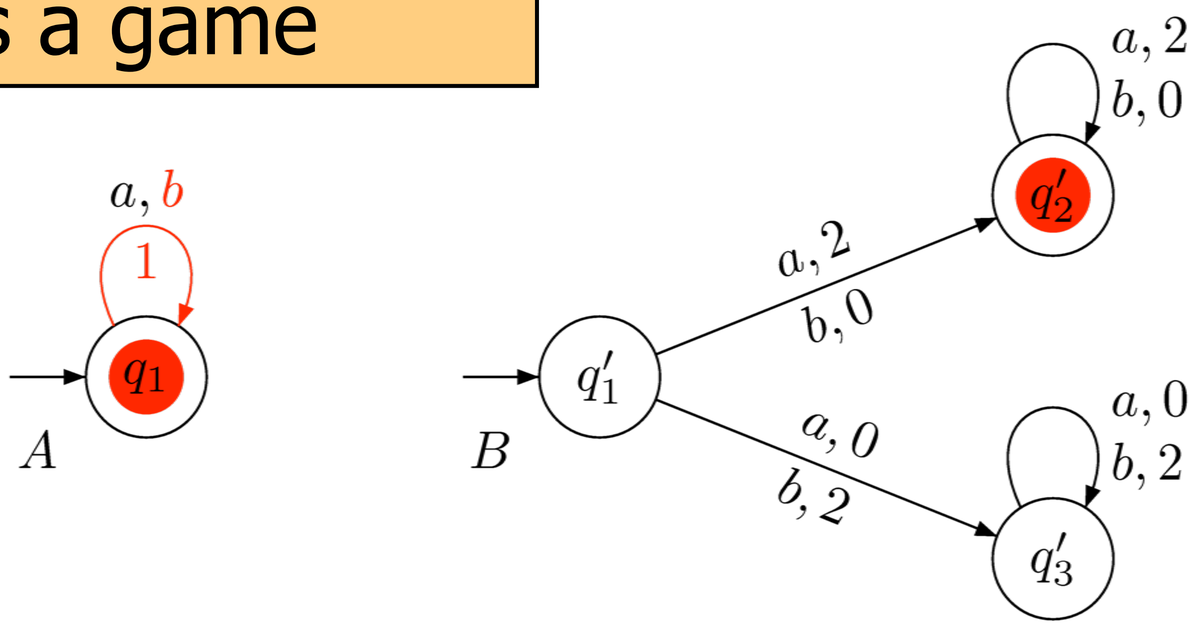


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1$

Simulator: $q'_1 \xrightarrow[2]{a} q'_2 \xrightarrow[0]{b} q'_2$

Language-inclusion game

Language inclusion as a game

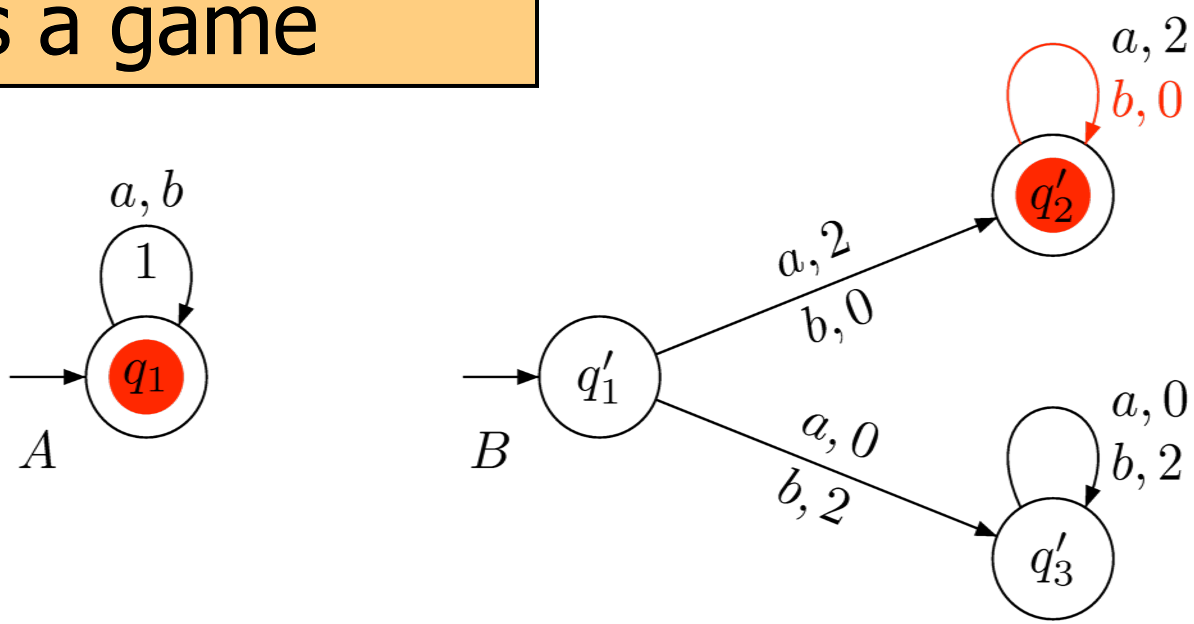


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1 \xrightarrow[1]{b} \cdots$

Simulator: $q'_1 \xrightarrow[2]{a} q'_2 \xrightarrow[0]{b} q'_2$

Language-inclusion game

Language inclusion as a game

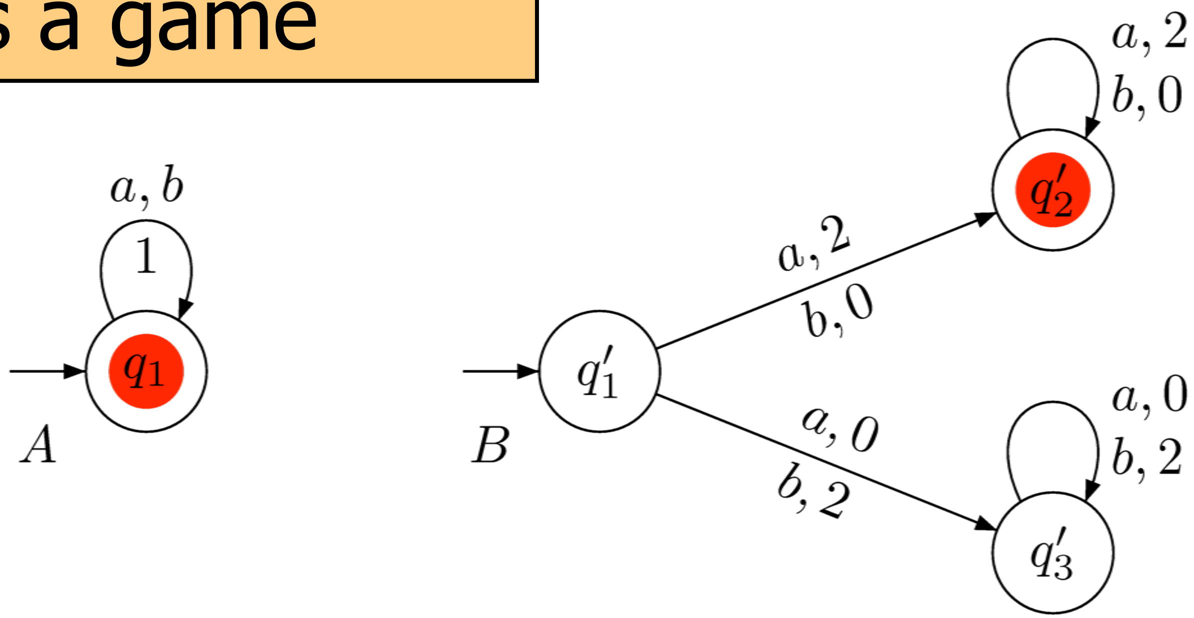


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1 \xrightarrow[1]{b} \cdots$

Simulator: $q'_1 \xrightarrow[2]{a} q'_2 \xrightarrow[0]{b} q'_2 \xrightarrow[0]{b} \cdots$

Language-inclusion game

Language inclusion as a game

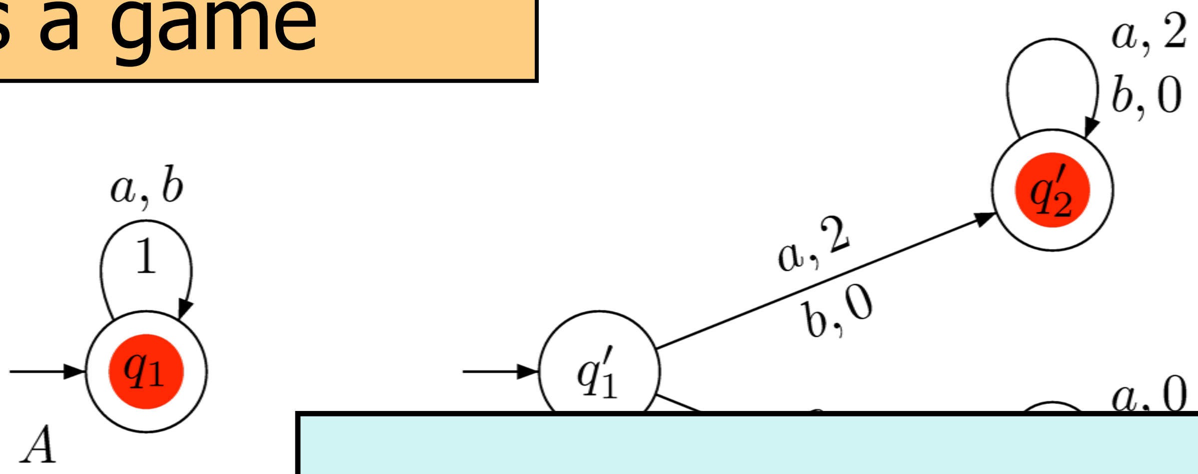


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1 \xrightarrow[1]{b} \cdots$ $\quad \mathsf{Disc}_{\frac{3}{4}}(1,1,1,\ldots) = \frac{1}{1-\frac{3}{4}} = 4.$

Simulator: $q'_1 \xrightarrow[2]{a} q'_2 \xrightarrow[0]{b} q'_2 \xrightarrow[0]{b} \cdots$ $\quad \mathsf{Disc}_{\frac{3}{4}}(2,0,0,\ldots) = 2.$

Language-inclusion game

Language inclusion as a game



Challenger wins since 4>2.

Challenger: $q_1 \xrightarrow[1]{a}$

However, $L_A(w) \leq L_B(w)$ for all w.

Simulator: $q_1' \xrightarrow[2]{a} q_2' \xrightarrow[0]{} q_2' \xrightarrow[0]{} \cdots$ $\quad \text{Disc}_{\frac{3}{4}}(2, 0, 0, \ldots) = 2.$

Language-inclusion game

The game is blind if the Challenger cannot observe the state of the Simulator.

Challenger has no winning strategy in the blind game
if and only if
$L_A(w) \leq L_B(w)$ for all words w.

Language-inclusion game

The game is blind if the Challenger cannot observe the state of the Simulator.

Challenger has no winning strategy in the blind game
if and only if
$L_A(w) \leq L_B(w)$ for all words w.

When the game is not blind, we say that B simulates A if the Challenger has no winning strategy.

Simulation implies language inclusion.

Simulation is decidable

	Quant. L. inclusion	Quant. simulation	(Reduction to)
Sup	PSpace	P	(weak parity)
LimSup	PSpace	NP $\cap$ coNP	(parity)
LimInf	PSpace	NP $\cap$ coNP	(parity)
LimAvg	?	NP $\cap$ coNP	(mean payoff)
Disc_λ	?	NP $\cap$ coNP	(discounted sum)

Universality and Equivalence

Universality problem:

Given $\nu \in \mathbb{Q}$, is $L_A(w) \geq \nu$ for all words w ?

Language equivalence problem:

Is $L_A(w) = L_B(w)$ for all words w ?

Complexity/decidability: same situation as Language inclusion.

Outline

- Motivation
- Weighted automata
- Decision problems
- Expressive power
- Closure properties

Reducibility

A class C of weighted automata can be reduced to a class C′ of weighted automata if

for all $A \in C$, there is $A' \in C'$ such that $L_A = L_{A'}$.

Reducibility

A class C of weighted automata can be reduced to a class C′ of weighted automata if

for all $A \in C$, there is $A' \in C'$ such that $L_A = L_{A'}$.

E.g. for boolean languages:

- Nondet. coBüchi can be reduced to nondet. Büchi
- Nondet. Büchi cannot be reduced to det. Büchi

  (nondet. Büchi cannot be determinized)

Some easy facts

Disc_λ and LimAvg can define quantitative languages with infinite range, Sup, LimInf and LimSup cannot.

Disc_λ and LimAvg cannot be reduced to Sup, LimInf and LimSup.

Some easy facts

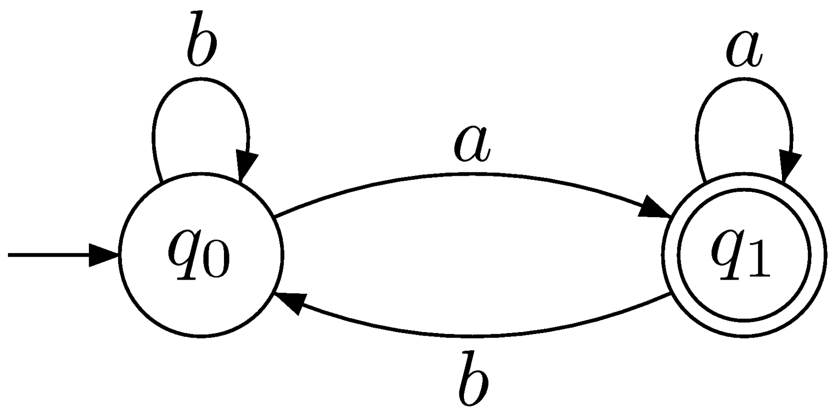
For discounted-sum, **prefixes** provide good approximations of the value.

For LimSup, LimInf and LimAvg, **suffixes** determine the value.

Disc_λ cannot be reduced to LimInf, LimSup and LimAvg.

LimInf, LimSup and LimAvg cannot be reduced to Disc_λ.

Büchi does not reduce to LimAvg



$$L_1 = (\Sigma^* \cdot a)^\omega$$

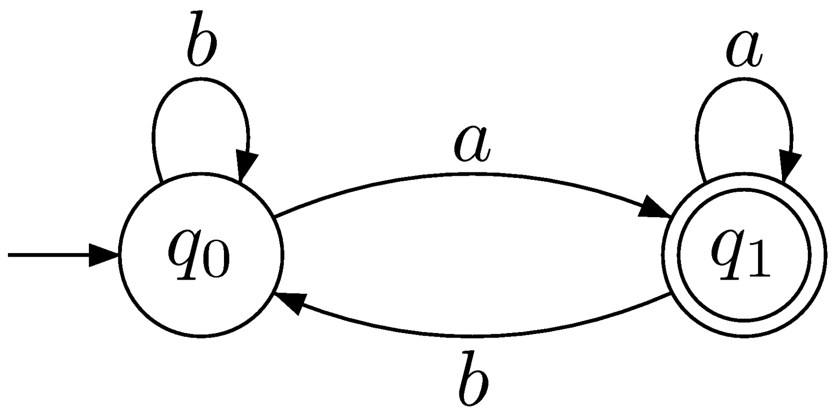
"infinitely many a"

Deterministic Büchi automaton

Assume that L_1 is definable by a LimAvg automaton A.

Then, all b-cycles in A have average weight ≤0.

Büchi does not reduce to LimAvg



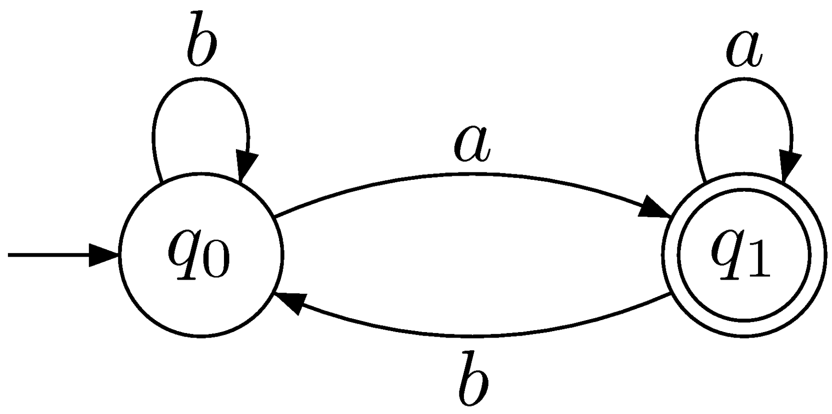
Deterministic Büchi automaton

$\mathsf{L}_1 = (\Sigma^* \cdot a)^\omega$

"infinitely many a"

Hence, the maximal average weight of a run over any word in $\Sigma^* \cdot b^n$ tends to (at most) 0 when $n \to \infty$.

Büchi does not reduce to LimAvg



Deterministic Büchi automaton

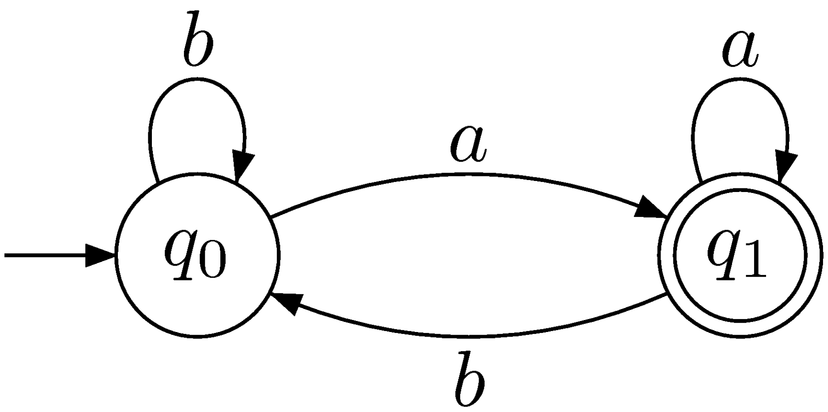
$\mathsf{L}_1 = (\Sigma^* \cdot a)^\omega$

"infinitely many a"

Let $w_n = (a \cdot b^n)^\omega$ We have $\mathsf{L}_1(w_n) = 1$

$w_n = \underbrace{a \cdot b \cdots b}_{v_n} \cdot \underbrace{a \cdot b \cdots b}_{v_n} \cdots$ where $v_n \leq \varepsilon$ for sufficiently large n.

Büchi does not reduce to LimAvg



Deterministic Büchi automaton

$\mathsf{L}_1 = (\Sigma^* \cdot a)^\omega$

"infinitely many a"

Let $w_n = (a \cdot b^n)^\omega$ We have $\mathsf{L}_1(w_n) = 1$

$w_n = \underbrace{a \cdot b \cdots b}_{v_n} \cdot a$

Hence, $\mathsf{LimAvg}(w_n) = 0 \neq 1.$

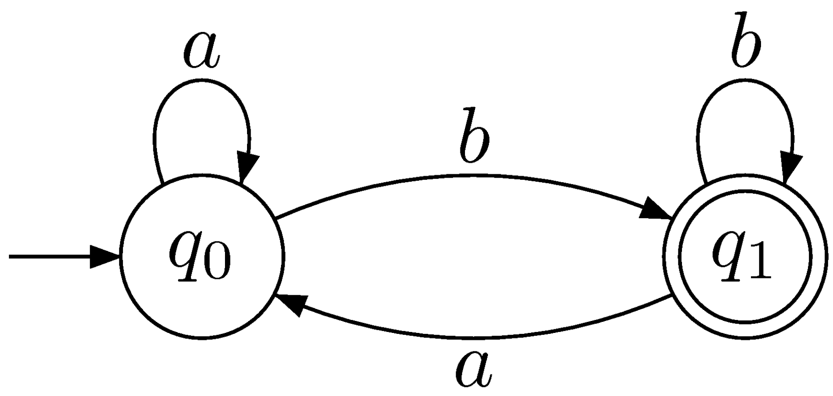
and A cannot exist !

(co)Büchi and LimAvg

det. Büchi cannot be reduced to LimAvg.

By analogous arguments,

det. coBüchi cannot be reduced to det. LimAvg.

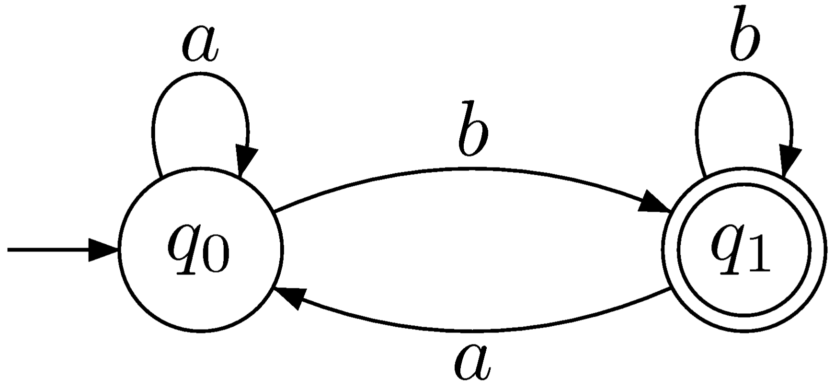


$\mathsf{L}_2 = \Sigma^* \cdot b^\omega$

"finitely many a"

Deterministic coBüchi automaton

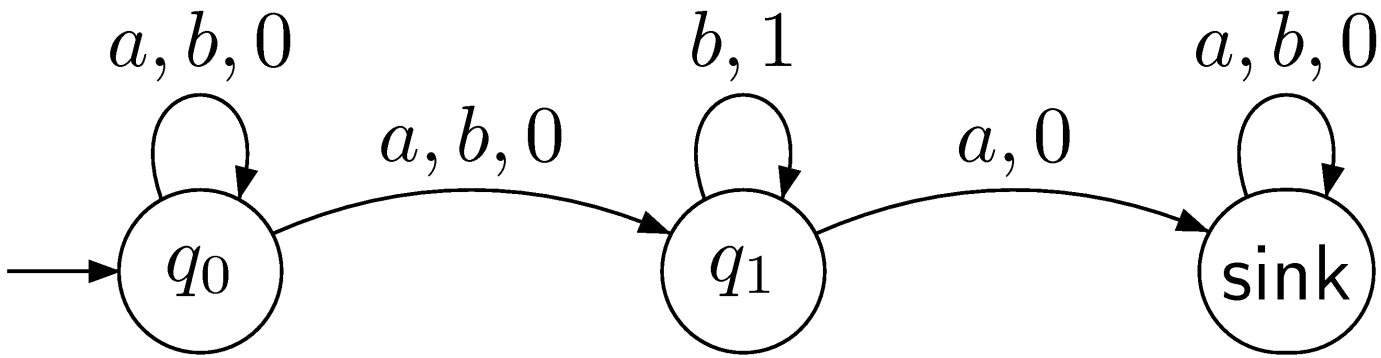
(co)Büchi and LimAvg



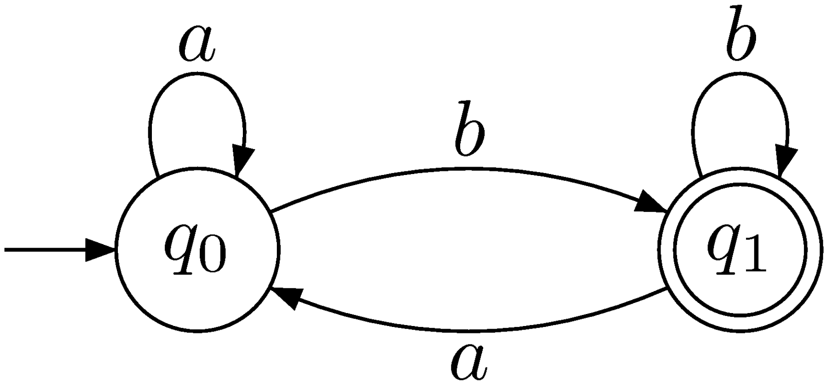
Det. coBüchi automaton

$L_2 = \Sigma^* \cdot b^\omega$

L_2 is defined by the following nondet. LimAvg automaton:



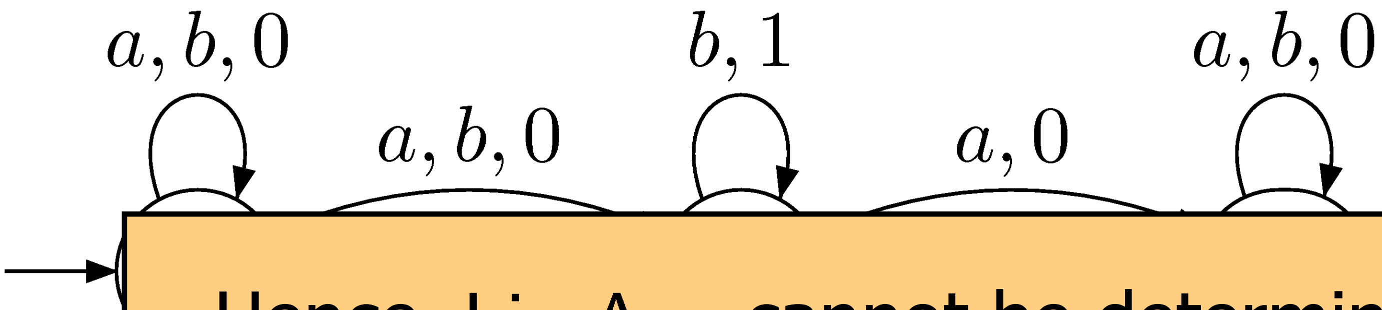
(co)Büchi and LimAvg



Det. coBüchi automaton

$L_2 = \Sigma^* \cdot b^\omega$

L_2 is defined by the following nondet. LimAvg automaton:

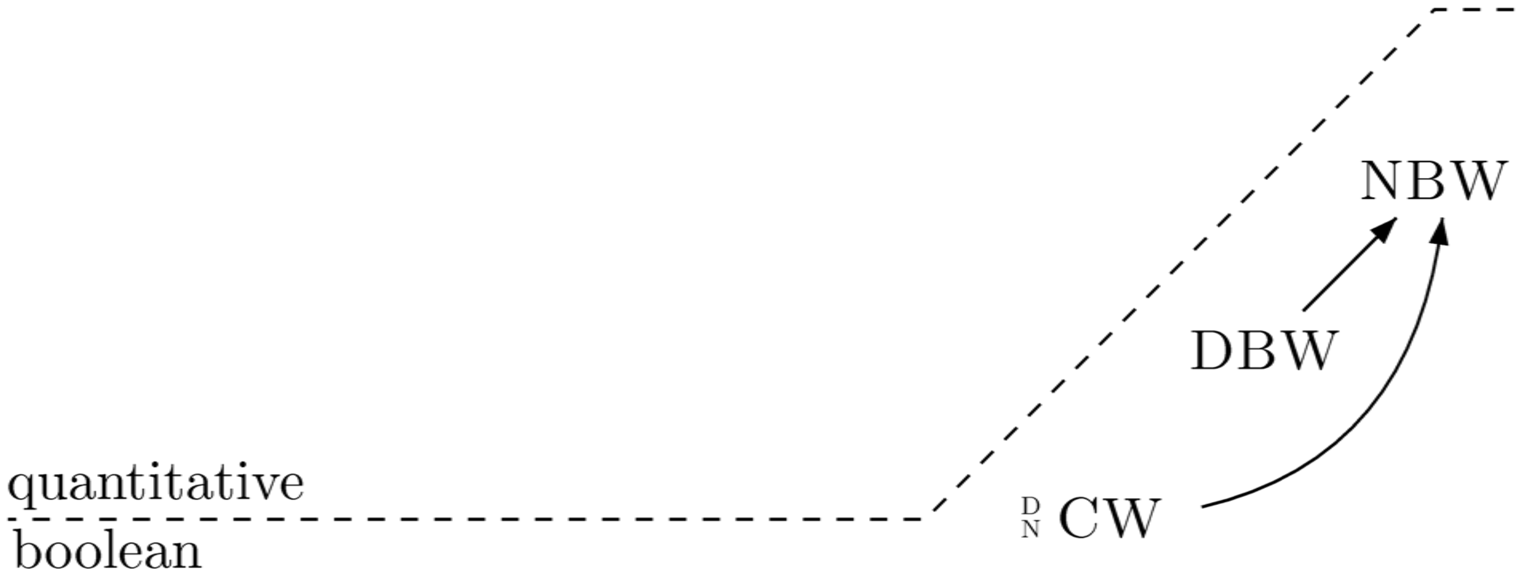


Hence, LimAvg cannot be determinized.

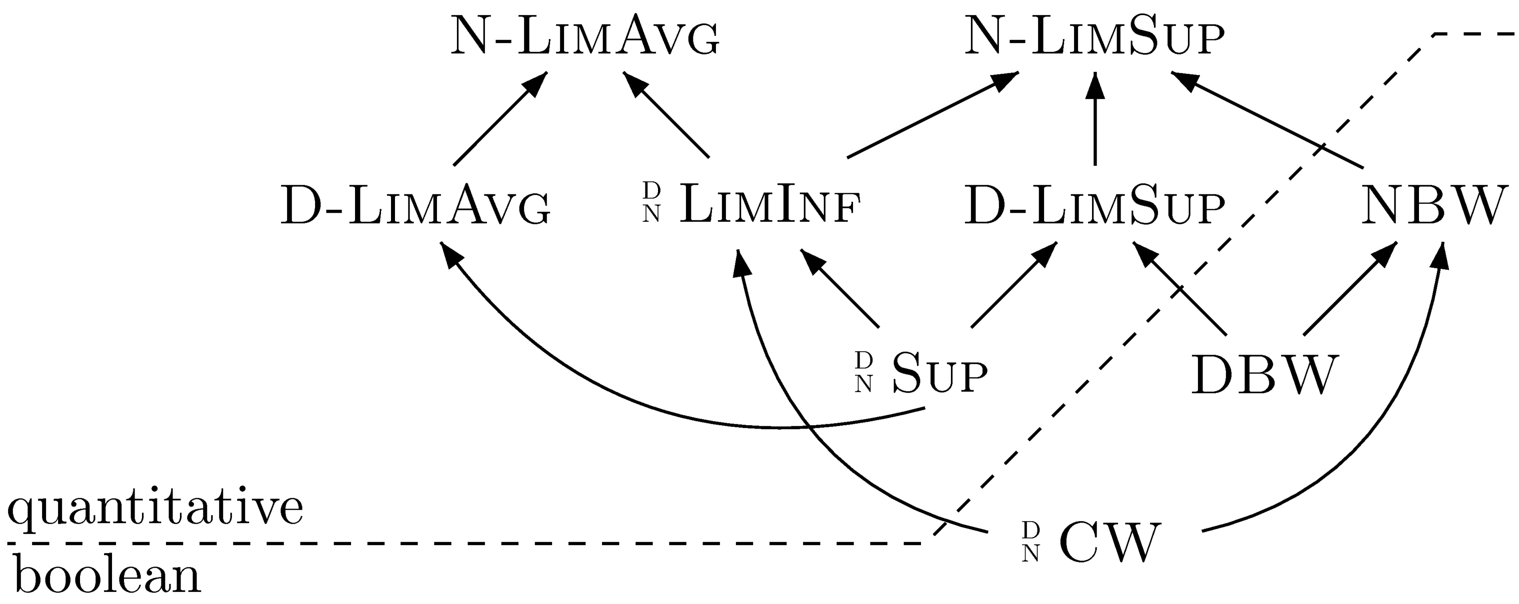
Reducibility relations

$$\begin{tikzcd}
 & & \mathrm{NBW} \\
 & \mathrm{DBW} \arrow[ur] & \\
{}^{\mathrm{D}}_{\mathrm{N}}\mathrm{CW} \arrow[uurr, bend right] & &
\end{tikzcd}$$

Reducibility relations



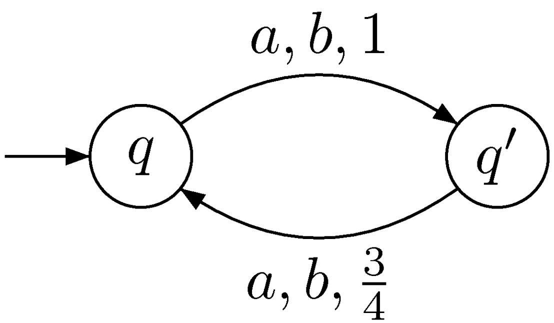
Reducibility relations



Expressive power of {0,1}-automata

$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

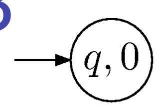
$\mathsf{Disc}_{[0,1]}$ is not reducible to $\mathsf{Disc}_{\{0,1\}}$.



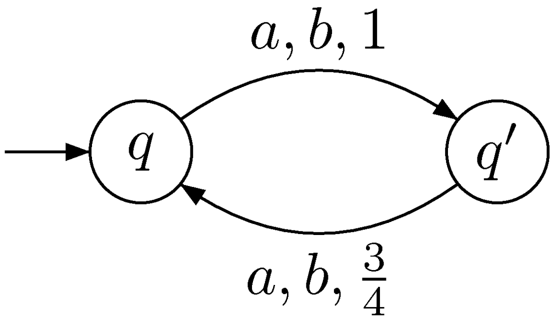
Expressive power of {0,1}-automata

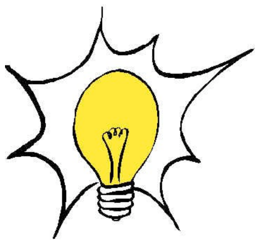
$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

B



A





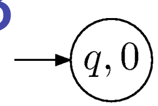
Store the value

$$\Delta_n = \sum_i^n v_i^A - \sum_i^n v_i^B$$

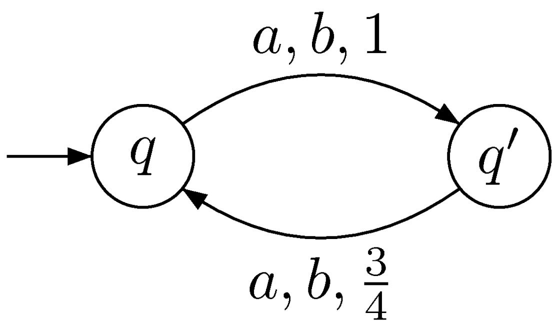
Expressive power of {0,1}-automata

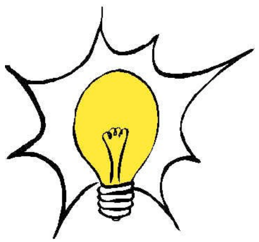
$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

B



A





Store the value

$$\Delta_n = \sum_i^n v_i^A - \sum_i^n v_i^B$$

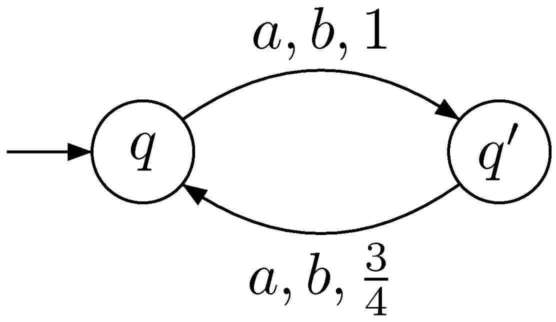
Key: Δ_n can take finitely many different values.

in the example, $\Delta_n \in \{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}\}$

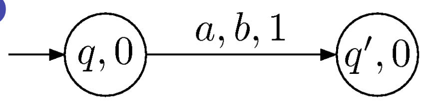
Expressive power of {0,1}-automata

$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

A



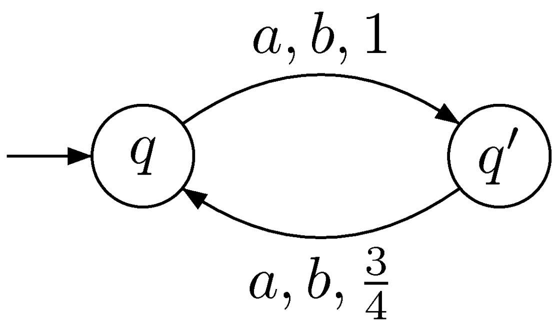
B



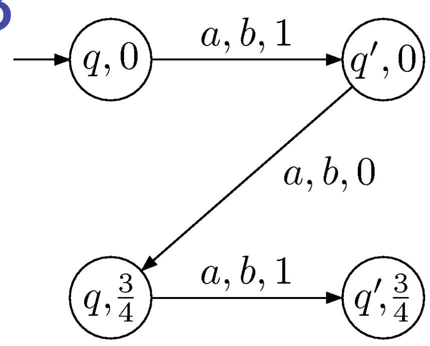
Expressive power of {0,1}-automata

$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

A



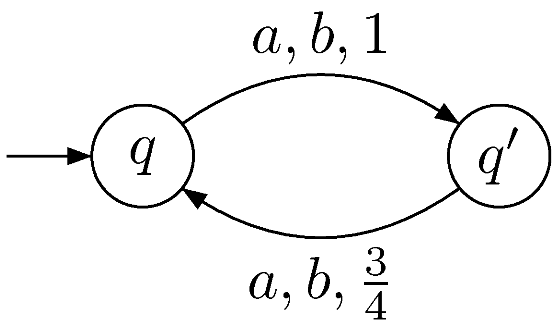
B



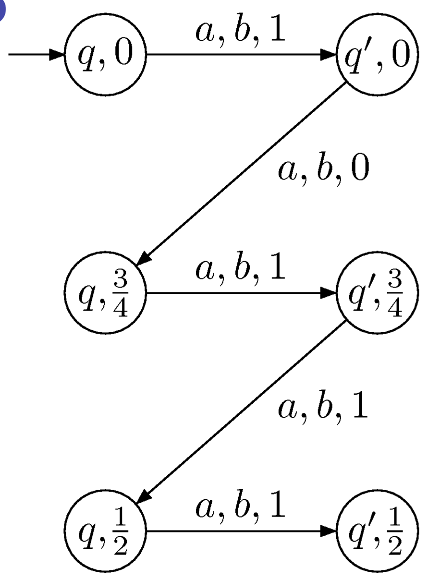
Expressive power of {0,1}-automata

$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

A

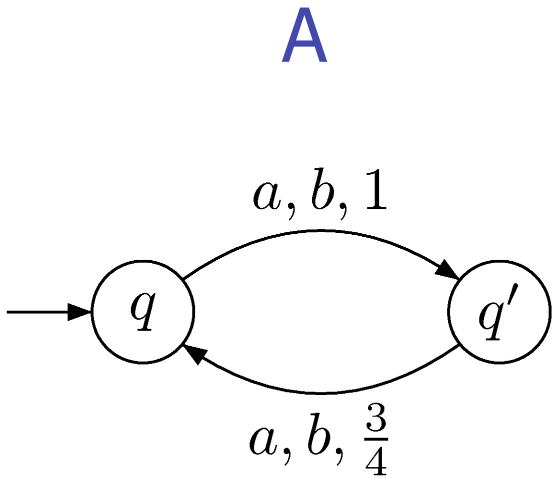


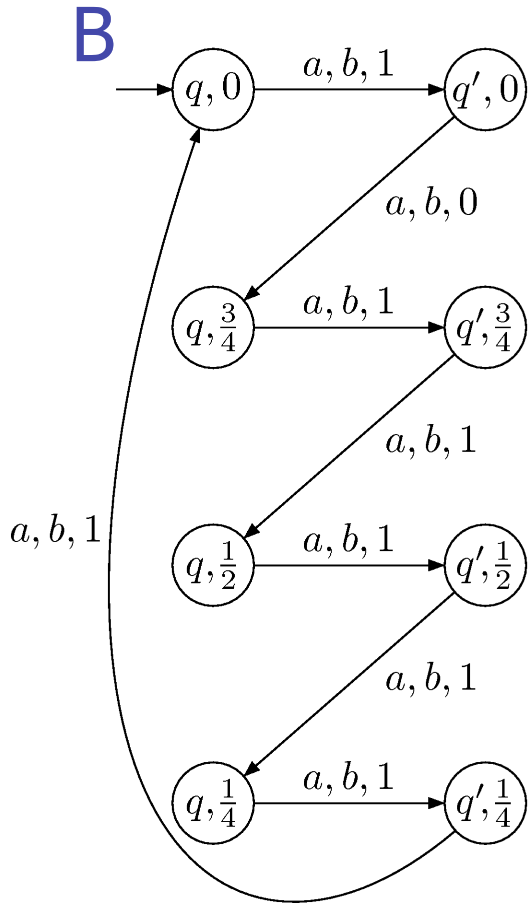
B



Expressive power of {0,1}-automata

$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

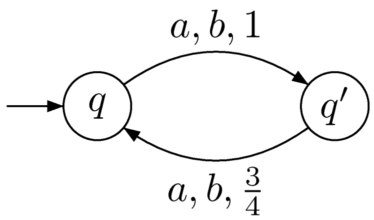




Expressive power of {0,1}-automata

$\mathsf{LimAvg}_{[0,1]}$ is reducible to $\mathsf{LimAvg}_{\{0,1\}}$.

A

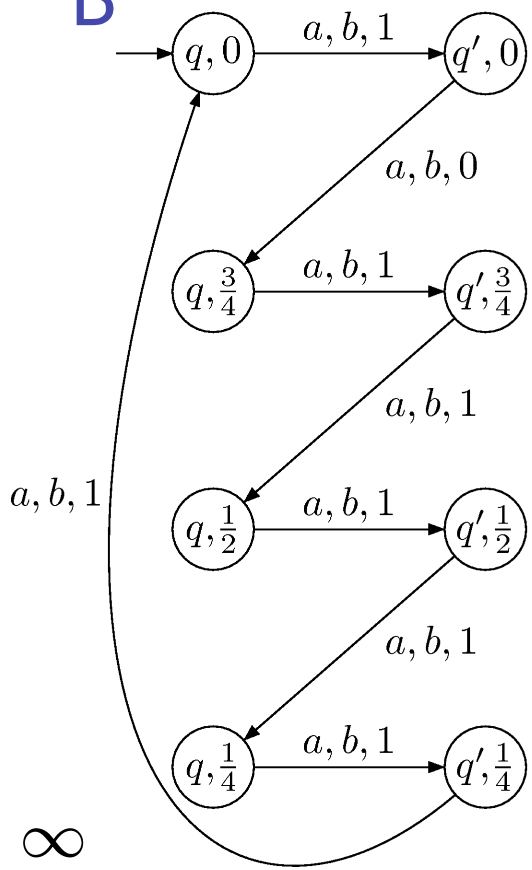


$$\Delta_n = \sum_i^n v_i^A - \sum_i^n v_i^B$$

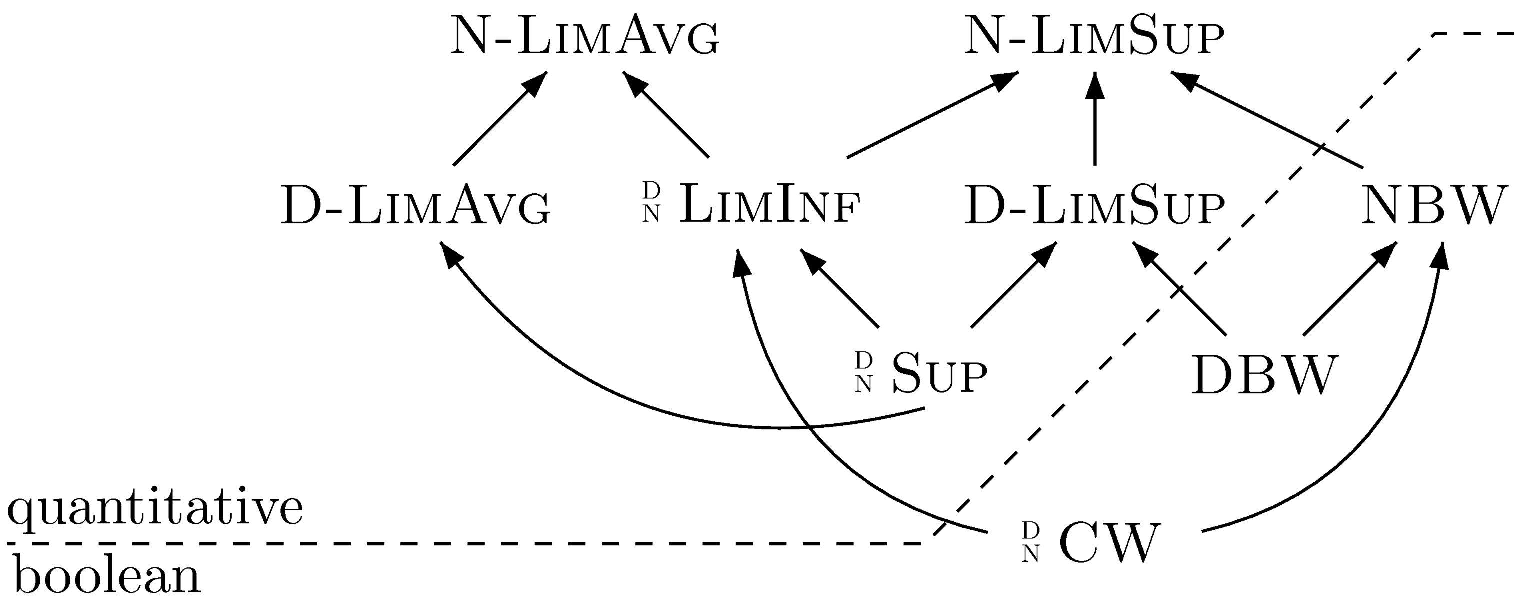
$0 \leq \Delta_n < 1$ for all $n \geq 0$

Therefore $\frac{\Delta_n}{n} \to 0$ for $n \to \infty$

B



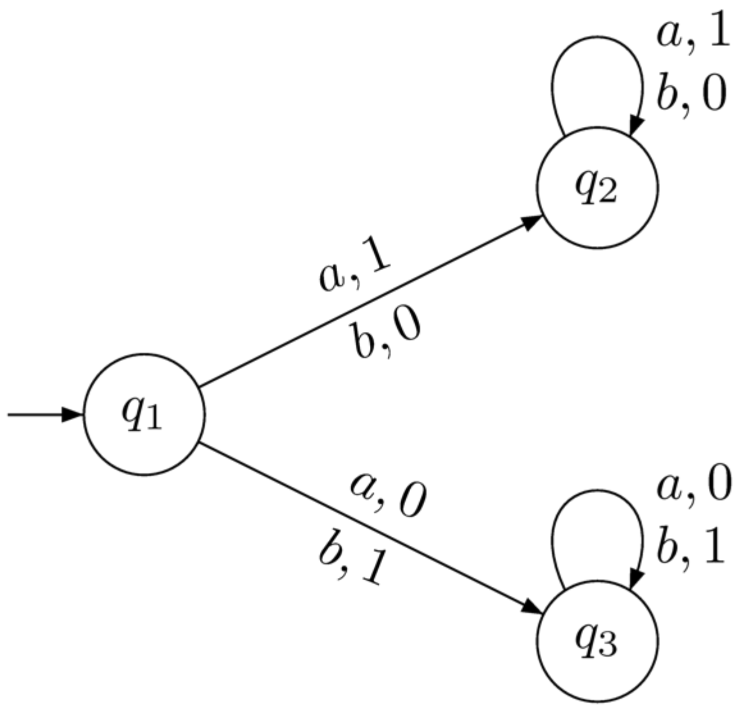
Reducibility relations



What about Discounted Sum ?

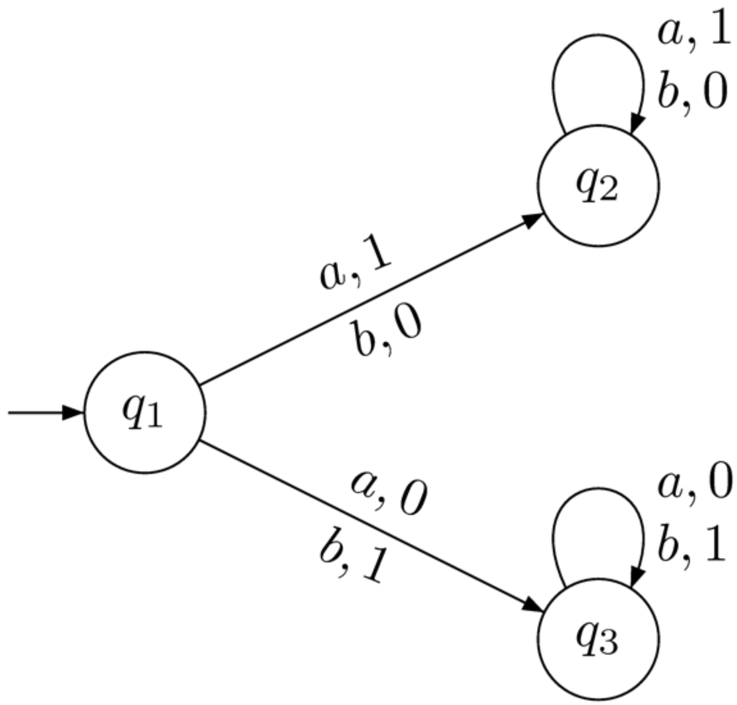
Last result

Disc_λ cannot be determinized.



$\lambda=3/4$

Disc$_\lambda$ cannot be determinized



Value of a word w :

$$\max(v_a(w), v_b(w))$$

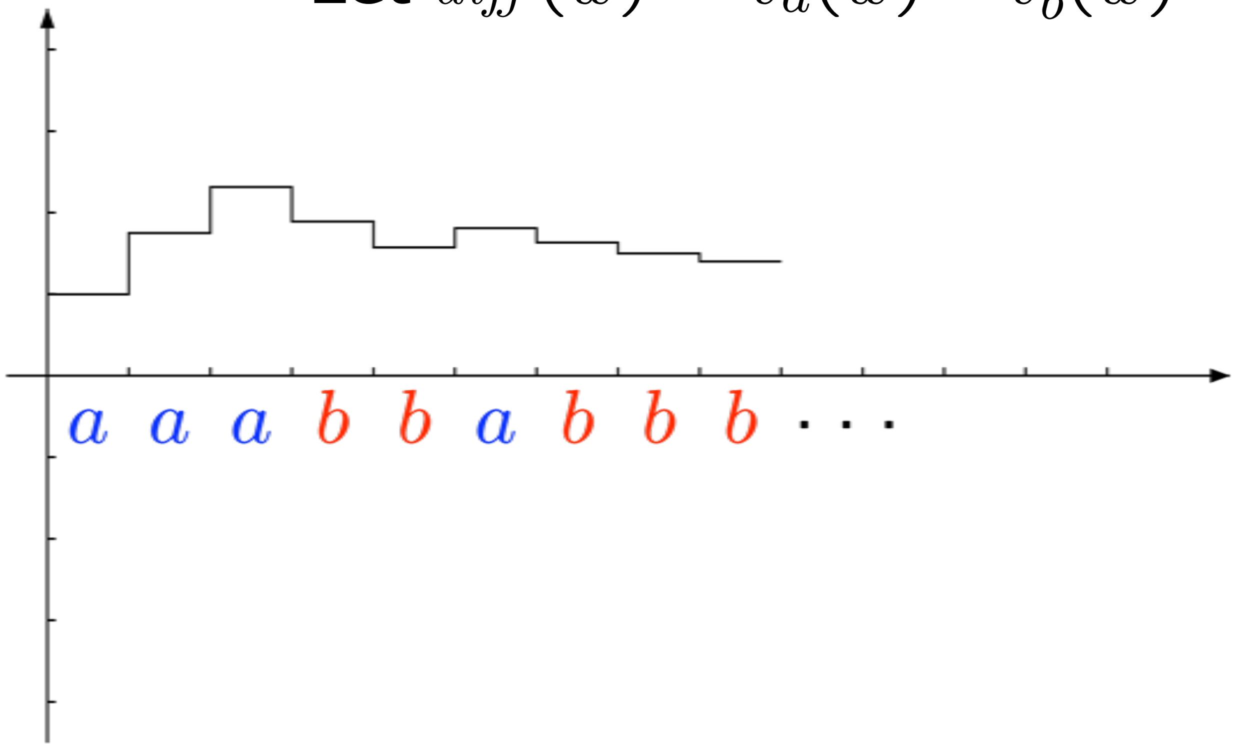
$$v_a(w) = \sum_{i \mid w_i = a}^{|w|} \lambda^i$$ disc. sum of a's

$$v_b(w) = \sum_{i \mid w_i = b}^{|w|} \lambda^i$$ disc. sum of b's

Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i|w_i=a}^{|w|} \lambda^i \qquad v_b(w) = \sum_{i|w_i=b}^{|w|} \lambda^i$$

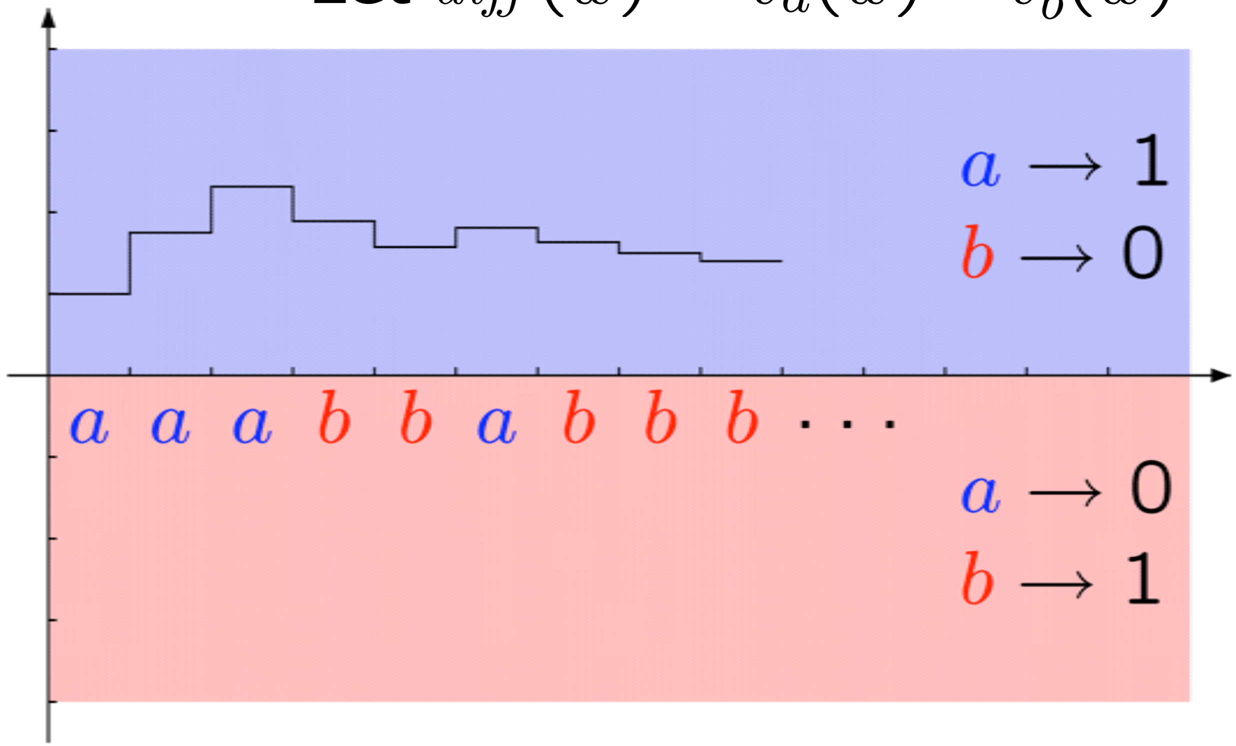
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i | w_i = a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i | w_i = b}^{|w|} \lambda^i$$

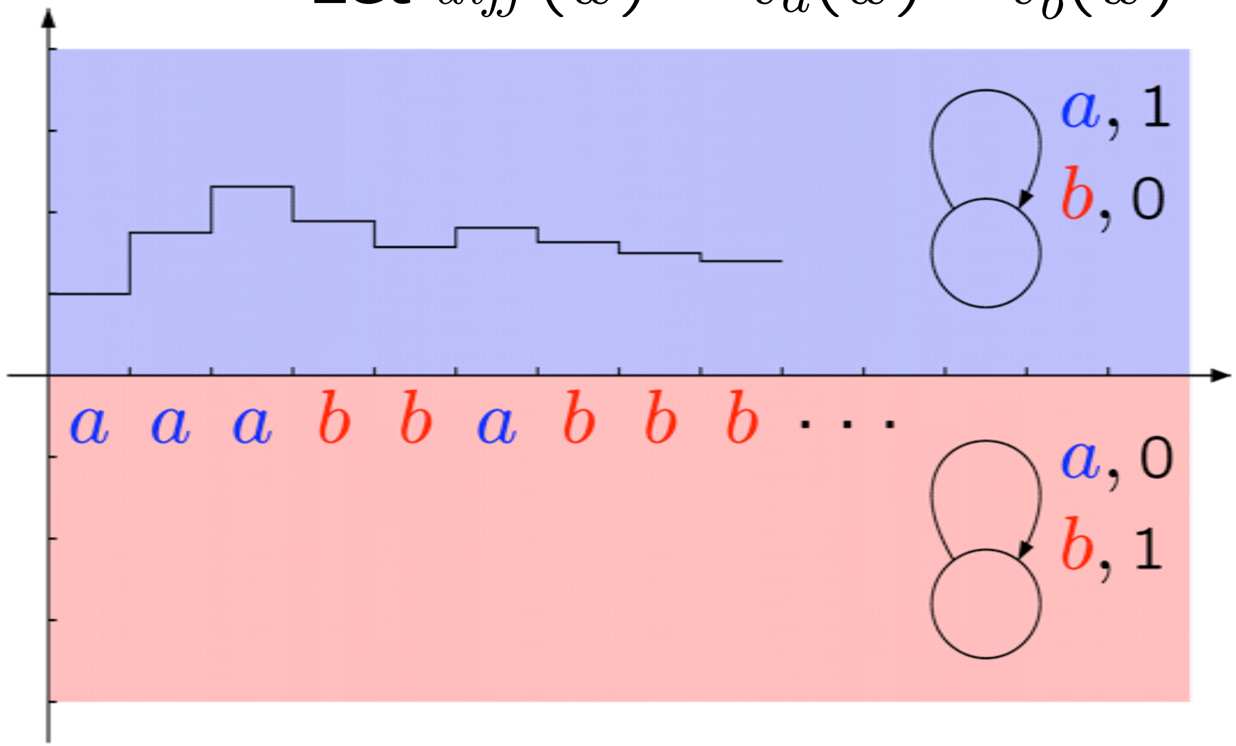
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i \mid w_i = a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i \mid w_i = b}^{|w|} \lambda^i$$

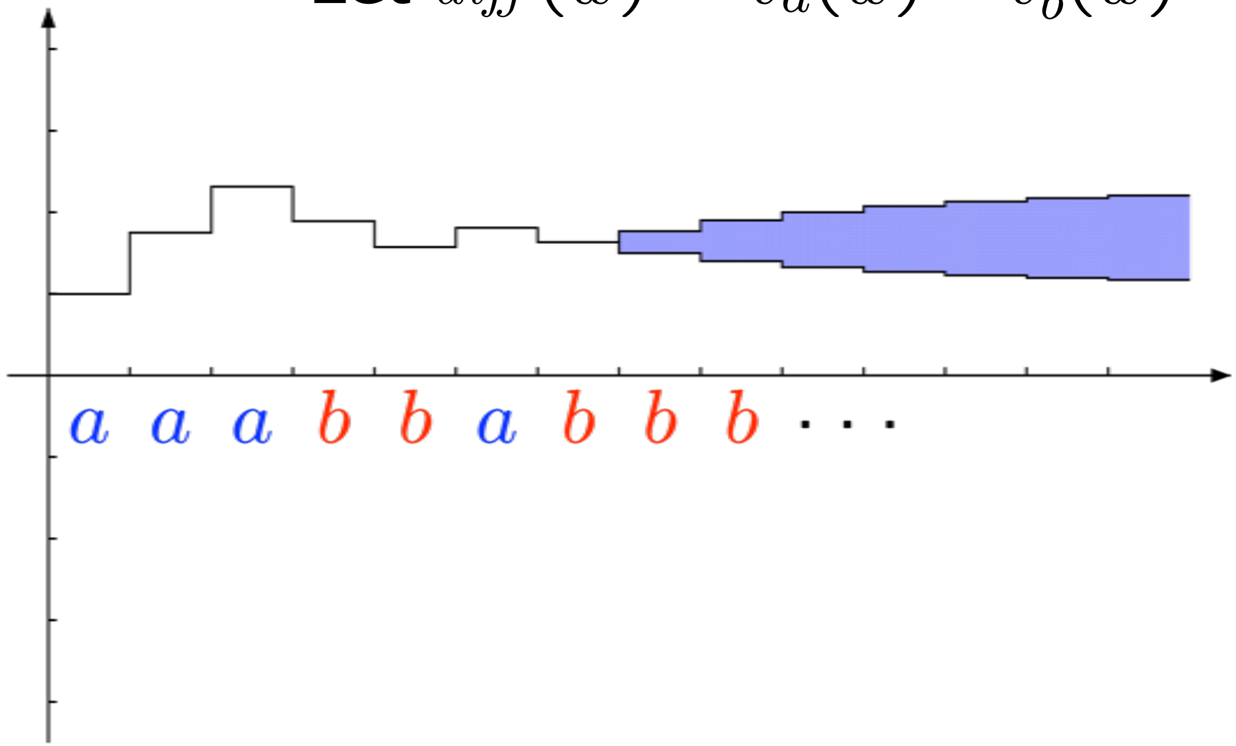
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i \mid w_i = a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i \mid w_i = b}^{|w|} \lambda^i$$

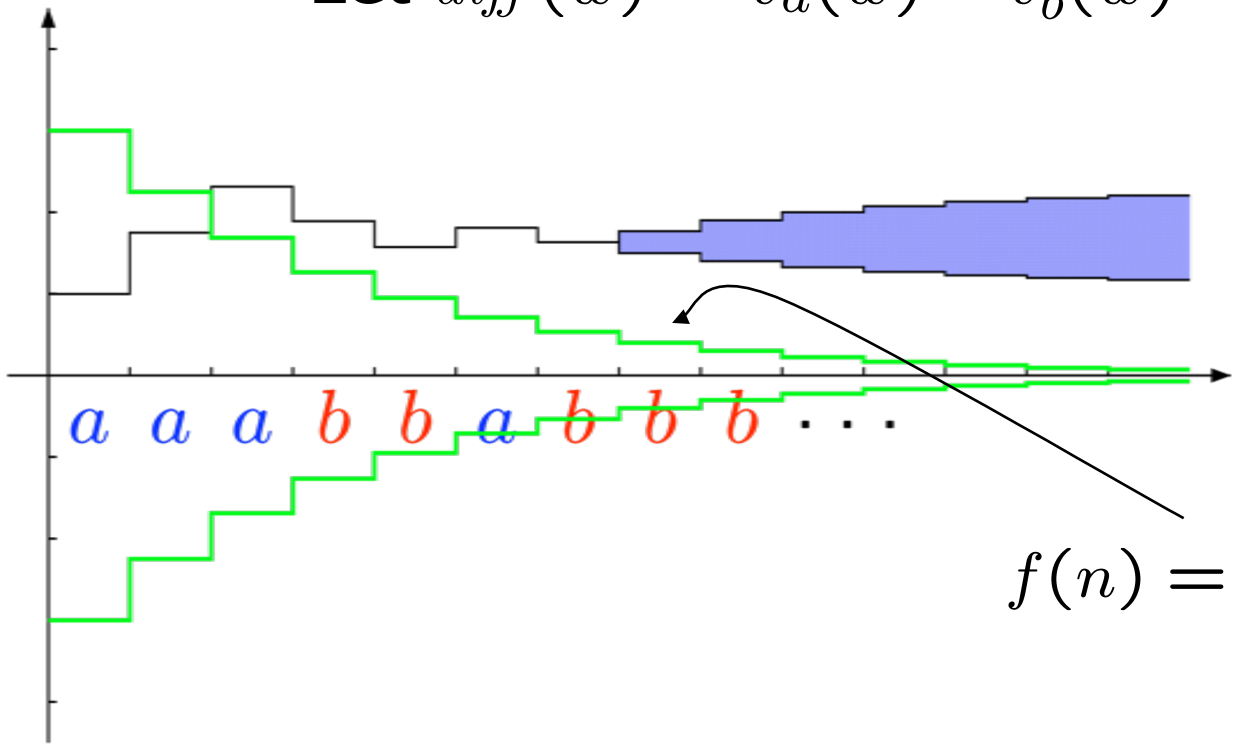
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i | w_i = a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i | w_i = b}^{|w|} \lambda^i$$

Let $\mathit{diff}(w) = v_a(w) - v_b(w)$

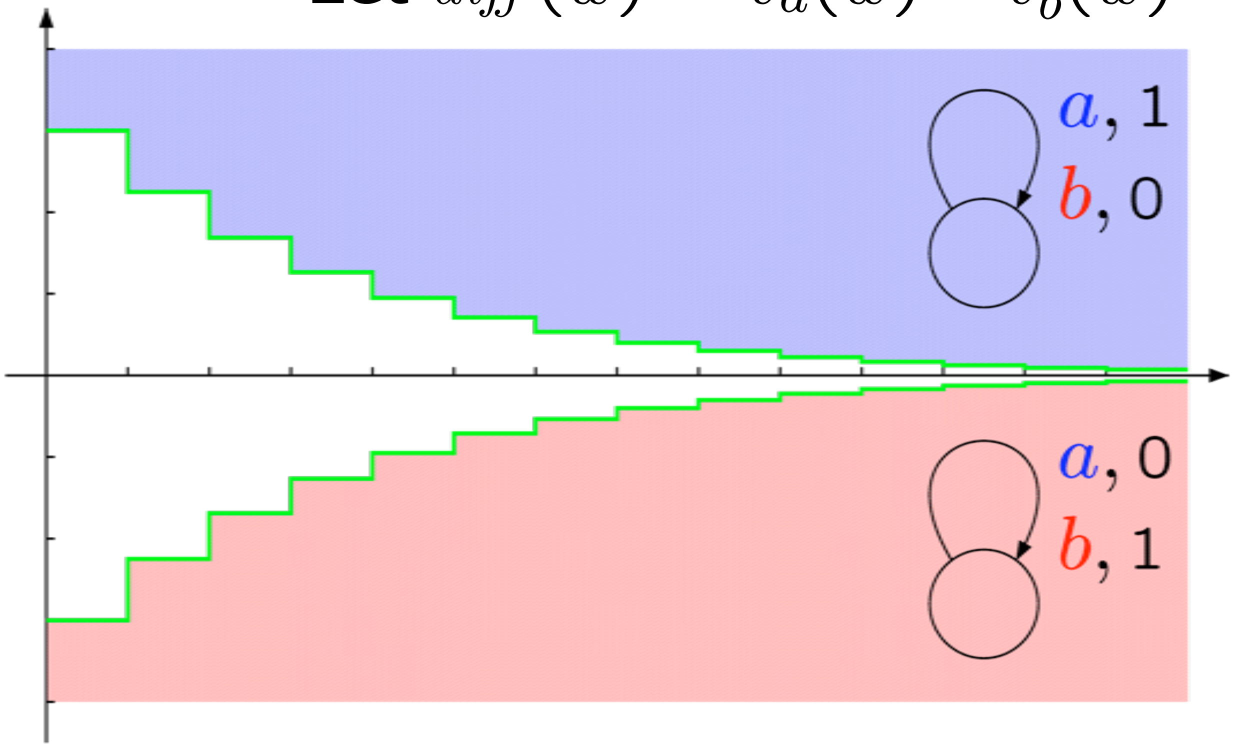


$$f(n) = \pm \frac{\lambda^n}{1-\lambda}$$

Disc_λ cannot be determinized

$$v_a(w) = \sum_{i \mid w_i = a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i \mid w_i = b}^{|w|} \lambda^i$$

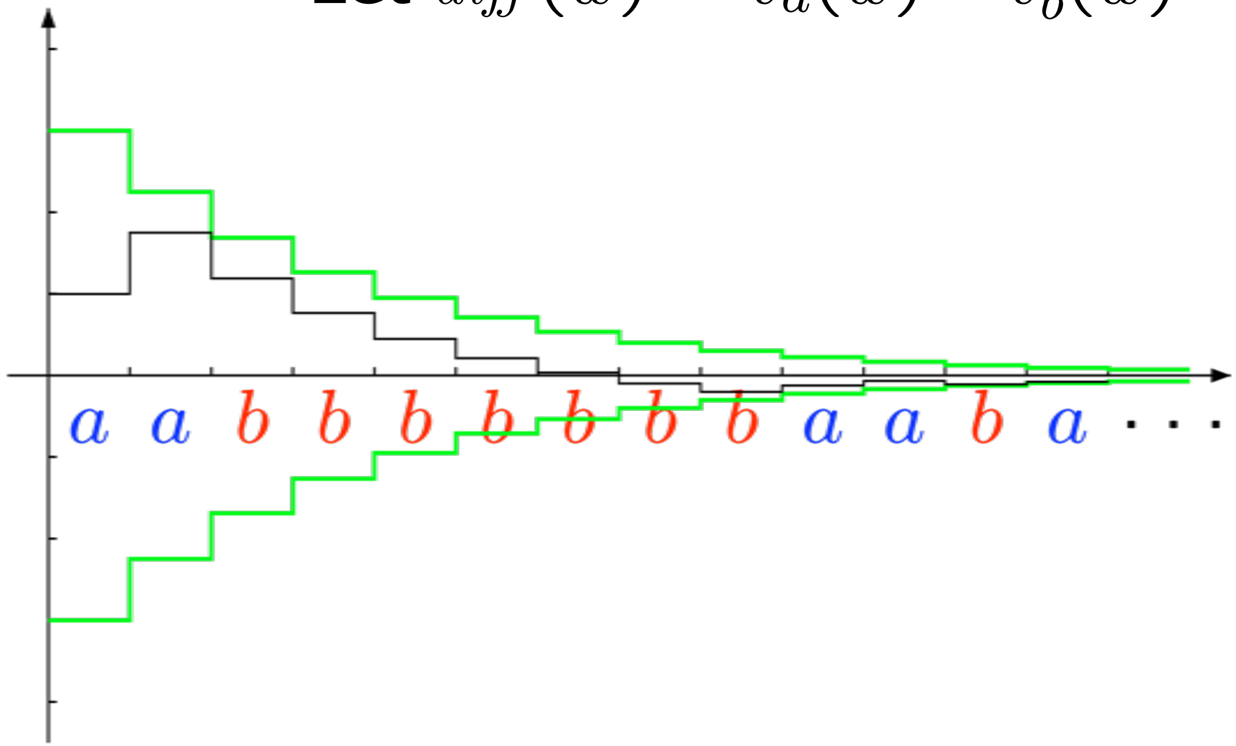
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i | w_i = a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i | w_i = b}^{|w|} \lambda^i$$

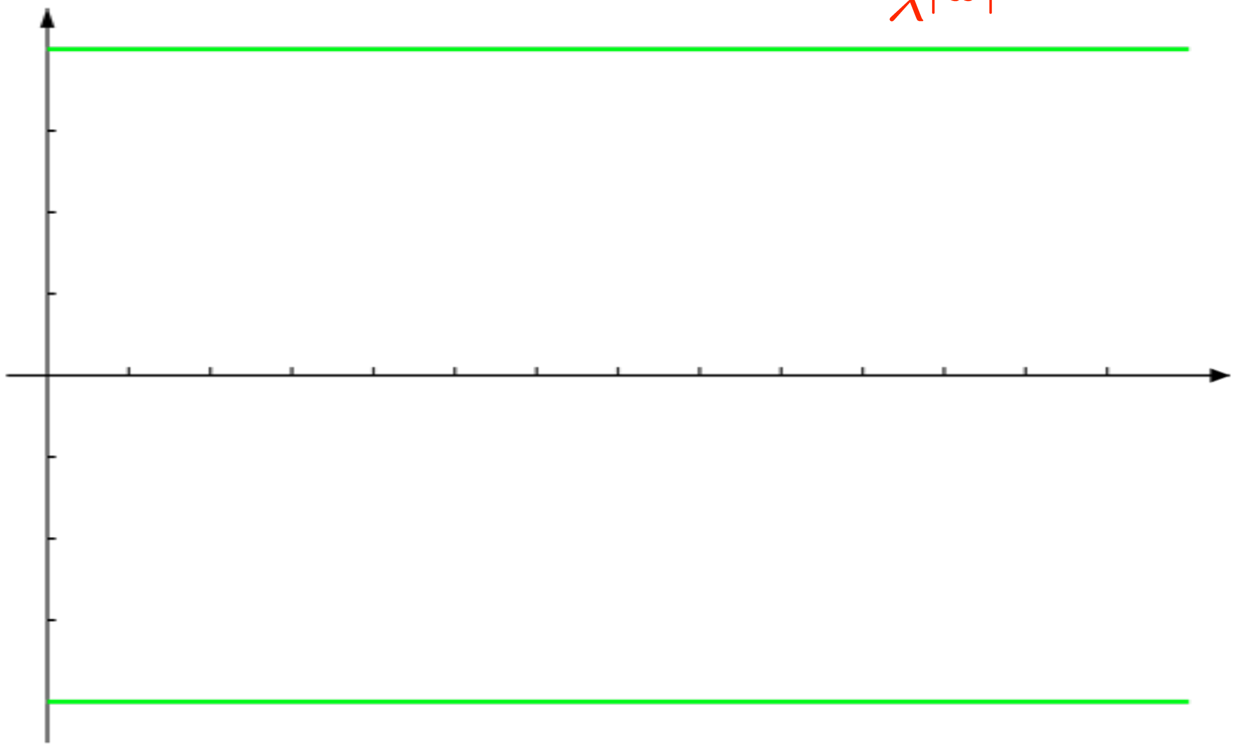
Let $\mathit{diff}\,(w) = v_a(w) - v_b(w)$



Disc_λ cannot be determinized

$$v_a(w) = \sum_{i|w_i=a}^{|w|} \lambda^i \qquad v_b(w) = \sum_{i|w_i=b}^{|w|} \lambda^i$$

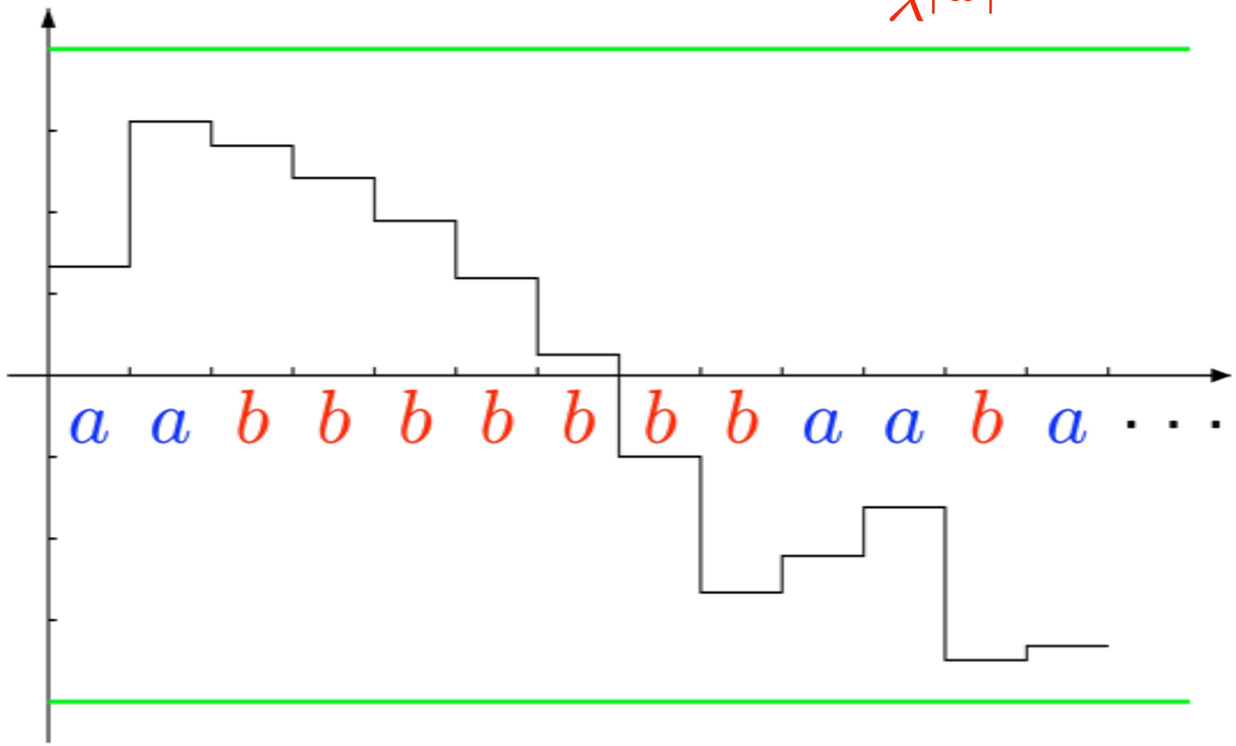
Let $\mathit{diff}(w) = \dfrac{v_a(w) - v_b(w)}{\lambda^{|w|}}$



Disc_λ cannot be determinized

$$v_a(w) = \sum_{i \mid w_i = a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i \mid w_i = b}^{|w|} \lambda^i$$

Let $\mathit{diff}\,(w) = \dfrac{v_a(w) - v_b(w)}{\lambda^{|w|}}$



Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i|w_i=a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i|w_i=b}^{|w|} \lambda^i$$

Let $\mathit{diff}(w) = \dfrac{v_a(w) - v_b(w)}{\lambda^{|w|}}$

If $\mathit{diff}(w) = s$

then $\begin{cases} \mathit{diff}(w \cdot a) & = & \frac{v_a(w)+\lambda^{|w|}-v_b(w)}{\lambda^{|w|+1}} & = & \frac{s+1}{\lambda} \\ \mathit{diff}(w \cdot b) & = & \frac{v_a(w)-v_b(w)-\lambda^{|w|}}{\lambda^{|w|+1}} & = & \frac{s-1}{\lambda} \end{cases}$

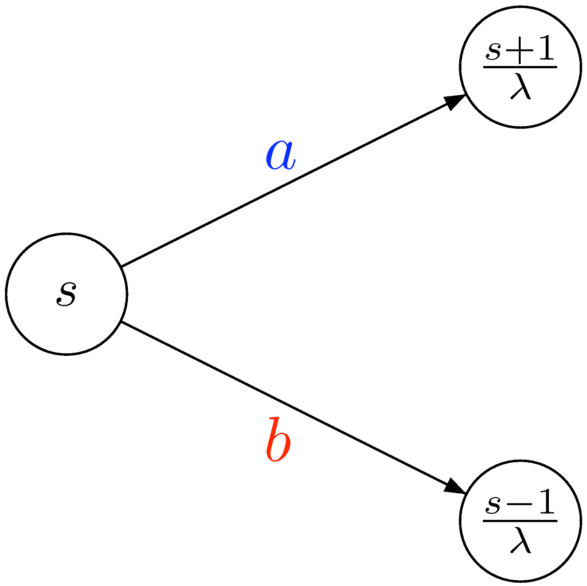
Disc$_\lambda$ cannot be determinized

$$v_a(w) = \sum_{i|w_i=a}^{|w|} \lambda^i \qquad\qquad v_b(w) = \sum_{i|w_i=b}^{|w|} \lambda^i$$

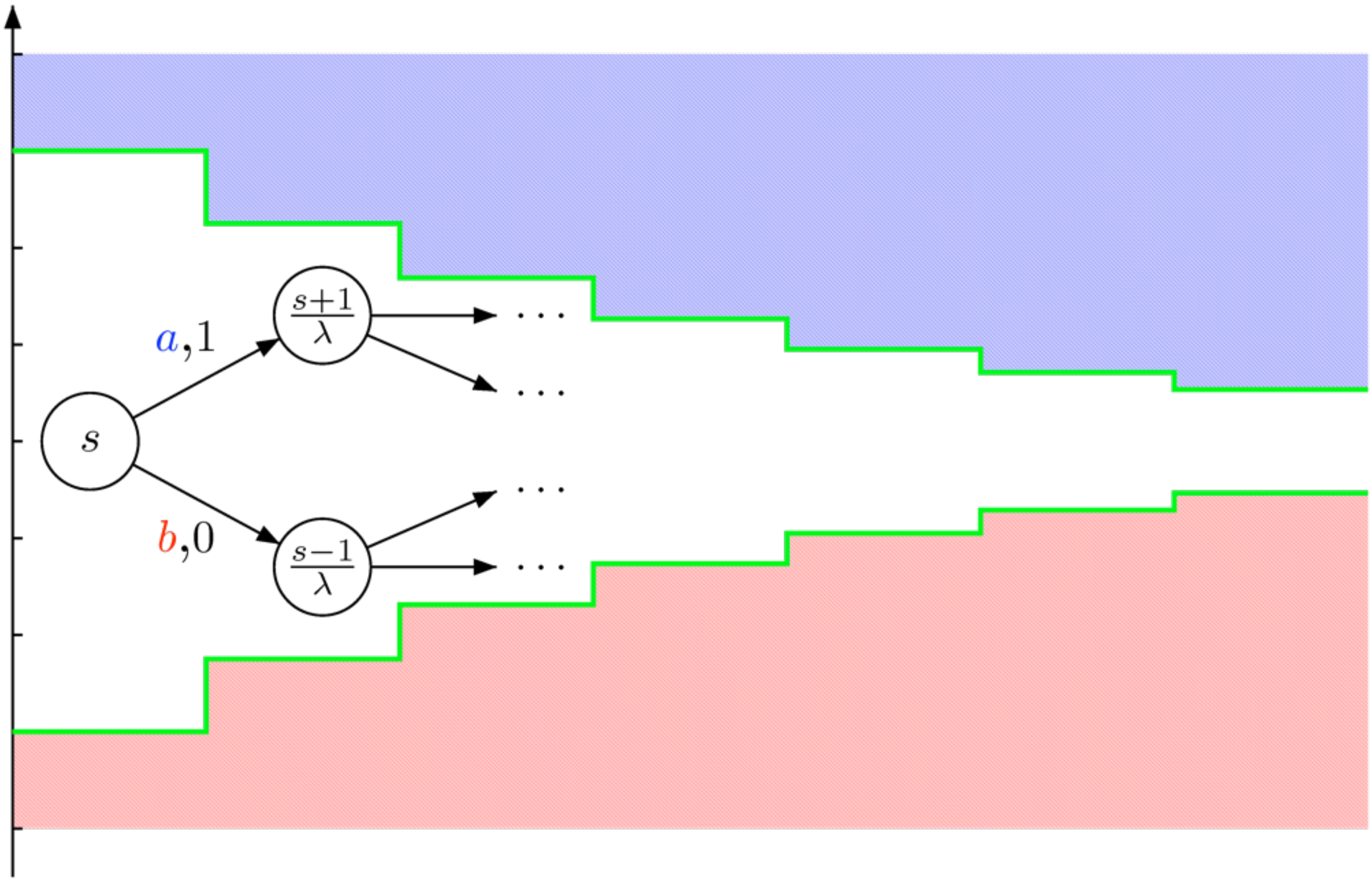
Let $\mathit{diff}\,(w) = \dfrac{v_a(w) - v_b(w)}{\lambda^{|w|}}$

If $\mathit{diff}\,(w) = s$

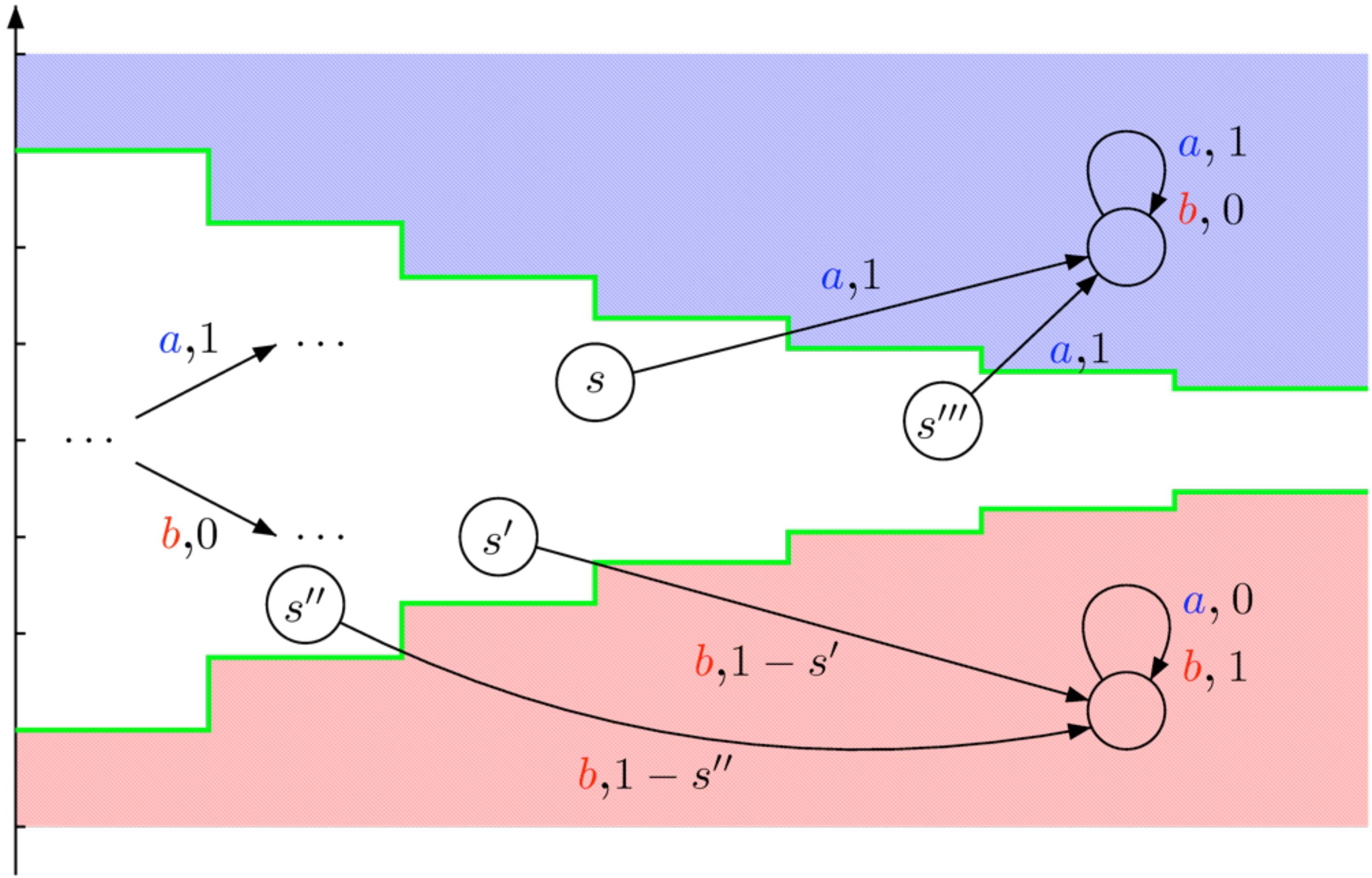
then $\begin{cases} \mathit{diff}\,(w \cdot a) & = & \frac{s+1}{\lambda} \\ \mathit{diff}\,(w \cdot b) & = & \frac{s-1}{\lambda} \end{cases}$



Disc_{λ} cannot be determinized



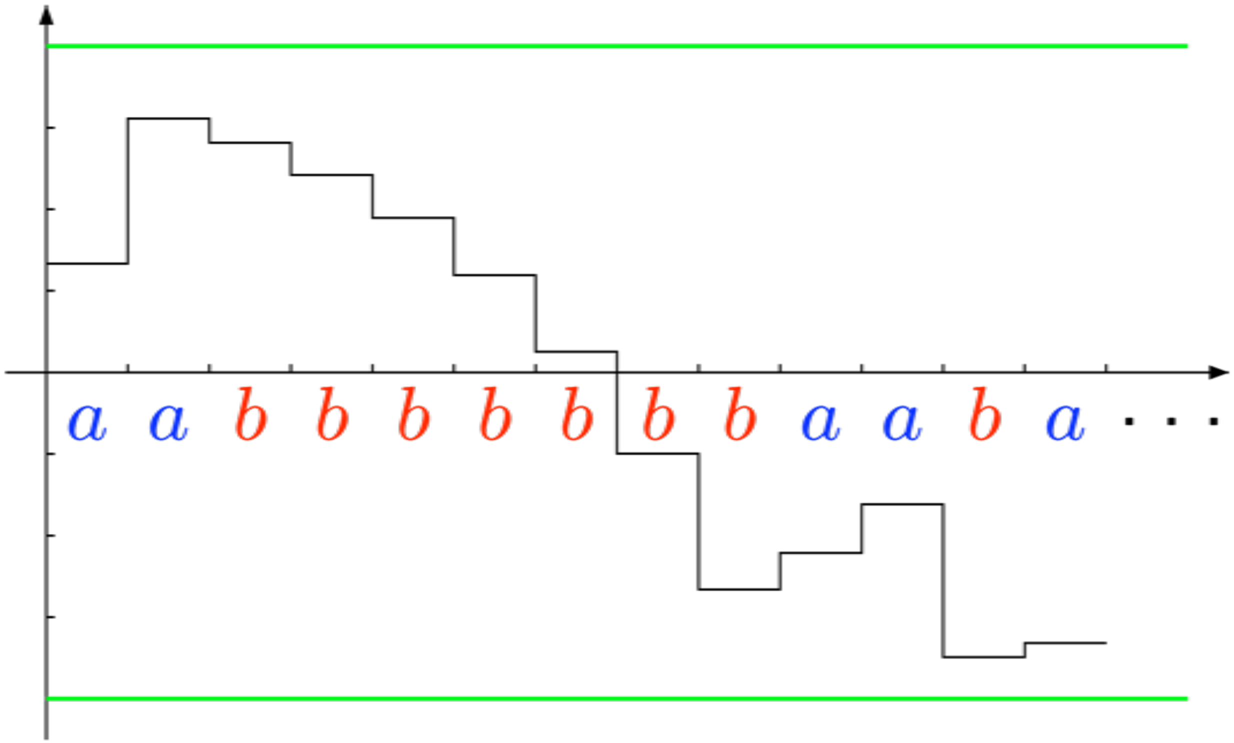
Disc$_\lambda$ cannot be determinized



Disc$_\lambda$ cannot be determinized

$$diff(w) = \frac{v_a(w) - v_b(w)}{\lambda^{|w|}}$$

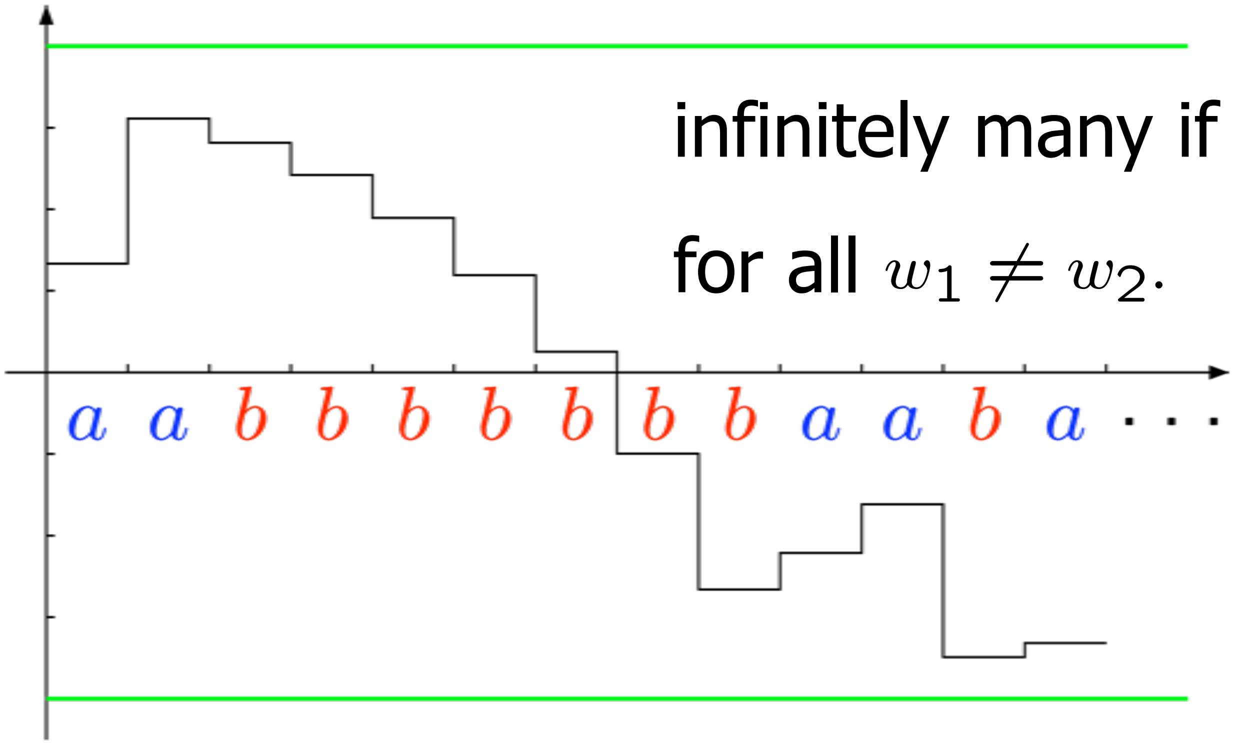
How many different values can $diff(w)$ take ?



Disc$_\lambda$ cannot be determinized

$$diff(w) = \frac{v_a(w) - v_b(w)}{\lambda^{|w|}}$$

How many different values can $diff(w)$ take ?



infinitely many if $diff(w_1) \neq diff(w_2)$

for all $w_1 \neq w_2$.

Disc_λ cannot be determinized

$$\mathit{diff}(w) = \frac{v_a(w) - v_b(w)}{\lambda^{|w|}}$$

How many different values can $\mathit{diff}(w)$ take ?

infinitely many if $\mathit{diff}(w_1) \neq \mathit{diff}(w_2)$

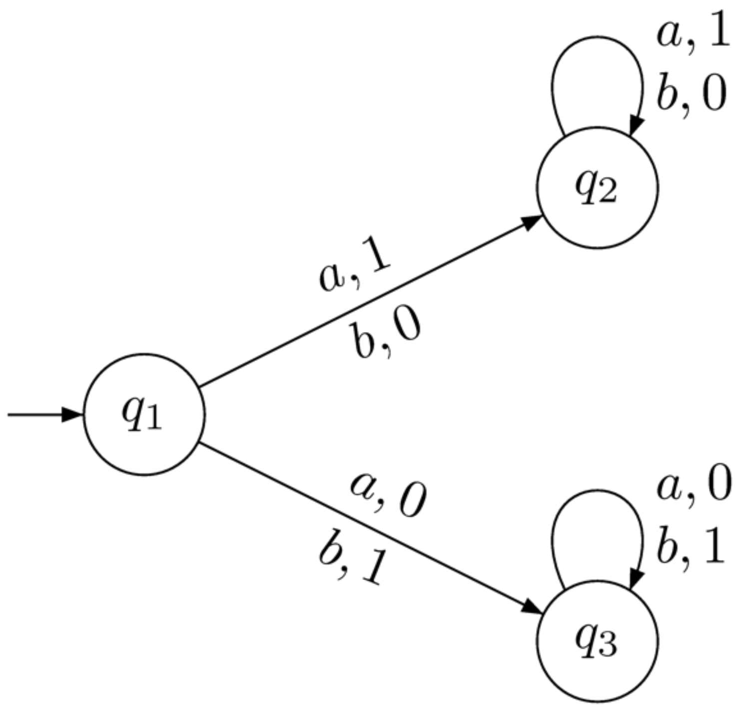
for all $w_1 \neq w_2$.

By a careful analysis of the shape of the family of equations,

it can be proven that no rational $\lambda \in]\frac{1}{2}, 1[$ can be a solution.

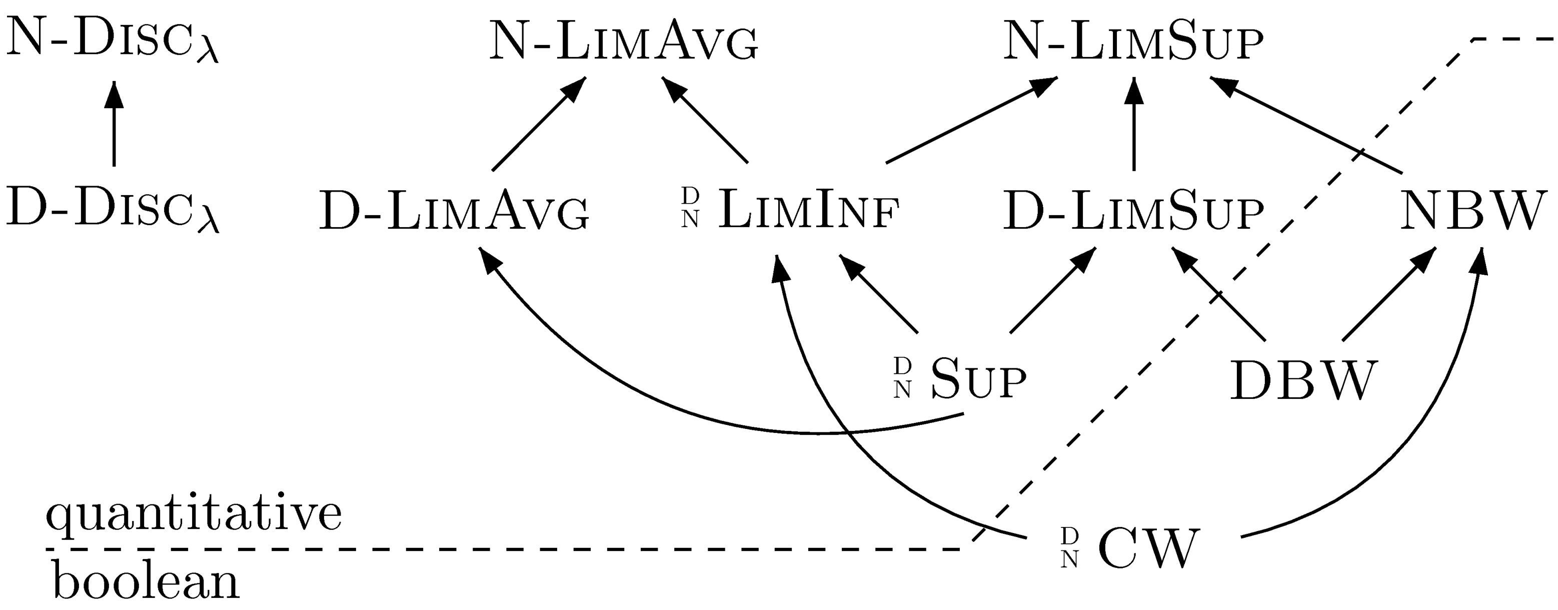
Last result

Disc_λ cannot be determinized.



λ=3/4

Reducibility relations



Outline

- Motivation
- Weighted automata
- Decision problems
- Expressive power
- Closure properties

Operations

$$\boxed{L_1, L_2 : \Sigma^\omega \longrightarrow \mathbb{R}}$$

Operations on quantitative languages:

- $\mathrm{shift}(L_1,c)(w) = L_1(w) + c$
- $\mathrm{scale}(L_1,c)(w) = c \cdot L_1(w)$ $\qquad (c>0)$

Operations

$$\boxed{L_1, L_2 : \Sigma^\omega \longrightarrow \mathbb{R}}$$

Operations on quantitative languages:

- $\text{shift}(L_1,c)(w) = L_1(w) + c$
- $\text{scale}(L_1,c)(w) = c \cdot L_1(w)$ $\quad (c>0)$
- $\max(L_1,L_2)(w) = \max(L_1(w),L_2(w))$ $\quad L_1 \cup L_2$
- $\min(L_1,L_2)(w) = \min(L_1(w),L_2(w))$ $\quad L_1 \cap L_2$
- $\text{complement}(L_1)(w) = 1-L_1(w)$ $\quad \Sigma^\omega \setminus L_1$

Operations

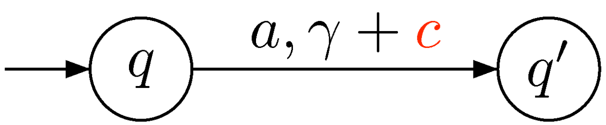
$$\boxed{L_1, L_2 : \Sigma^\omega \rightarrow \mathbb{R}}$$

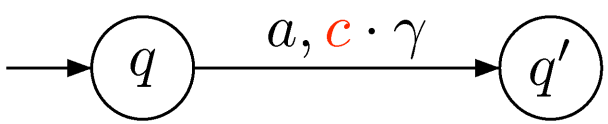
Operations on quantitative languages:

- $\text{shift}(L_1,c)(w) = L_1(w) + c$
- $\text{scale}(L_1,c)(w) = c \cdot L_1(w)$ $(c>0)$
- $\max(L_1,L_2)(w) = \max(L_1(w),L_2(w))$ $L_1 \cup L_2$
- $\min(L_1,L_2)(w) = \min(L_1(w),L_2(w))$ $L_1 \cap L_2$
- $\text{complement}(L_1)(w) = 1-L_1(w)$ $\Sigma^\omega \setminus L_1$
- $\text{sum}(L_1,L_2)(w) = L_1(w) + L_2(w)$

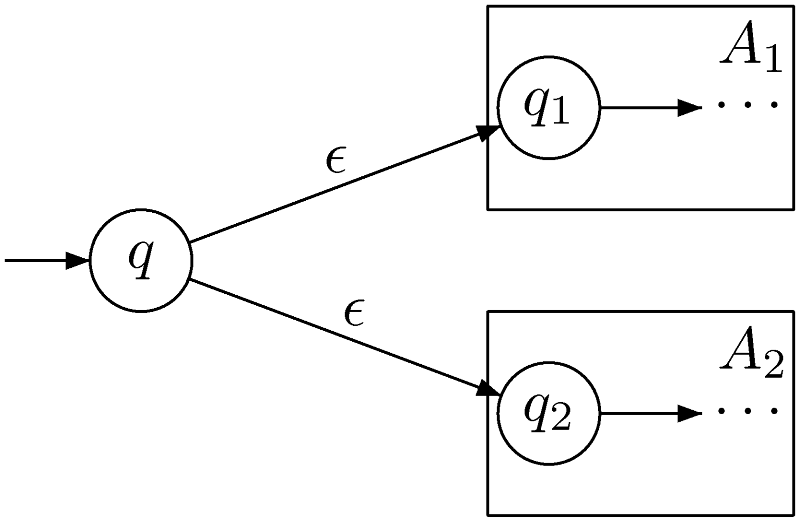
Closure properties

All classes of weighted automata are closed under **shift** and **scale**.





All classes of nondeterministic weighted automata are closed under **max**.



Closure properties

	$\cup$ max	$\cap$ min	$\Sigma^\omega \setminus L$ comp.	sum
$^{\mathrm{D}}_{\mathrm{N}}$ Sup	$\surd$			
$^{\mathrm{D}}_{\mathrm{N}}$ LimInf	$\surd$			
DLimSup	$\surd$			
NLimSup	$\surd$			
DLimAvg	$\times$			
NLimAvg	$\surd$			
DDisc	$\times$			
NDisc	$\surd$			

Closure properties

	$\cup$	$\cap$	$\Sigma^\omega \setminus L$	
	max	min	comp.	sum
${}^{\mathrm{D}}_{\mathrm{N}}\textsc{Sup}$	$\surd$	$\surd$	$\times$	$\surd$
${}^{\mathrm{D}}_{\mathrm{N}}\textsc{LimInf}$	$\surd$	$\surd$	$\times$	$\surd$
$\textsc{DLimSup}$	$\surd$	$\surd$	$\times$	$\surd$
$\textsc{NLimSup}$	$\surd$	$\surd$	$\surd$	$\surd$
$\textsc{DLimAvg}$	$\times$			
$\textsc{NLimAvg}$	$\surd$			
$\textsc{DDisc}$	$\times$			
$\textsc{NDisc}$	$\surd$			

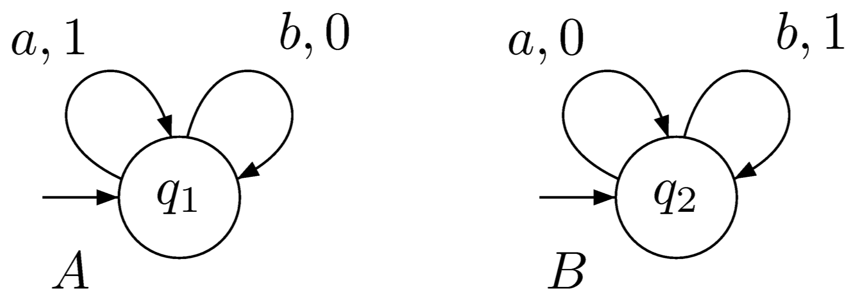
Analogous results for boolean languages.

Closure properties

There is no nondeterministic LimAvg automaton for the language $L_m = \min(L_a, L_b)$.

Closure properties

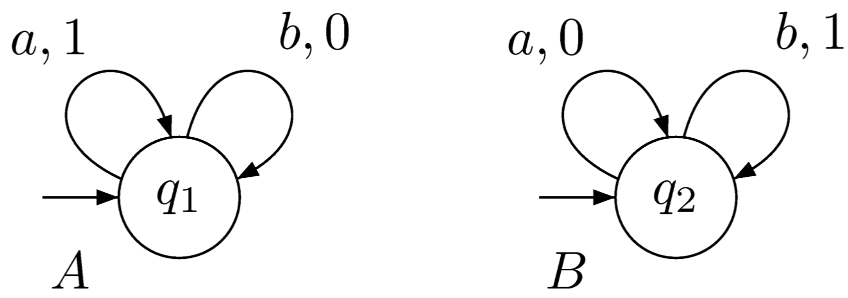
There is no nondeterministic LimAvg automaton for the language $L_m = \min(L_a, L_b)$.



Assume that $L_m = \min(L_a, L_b)$ is definable by a LimAvg automaton C.

Closure properties

There is no nondeterministic LimAvg automaton for the language $L_m = \min(L_a, L_b)$.



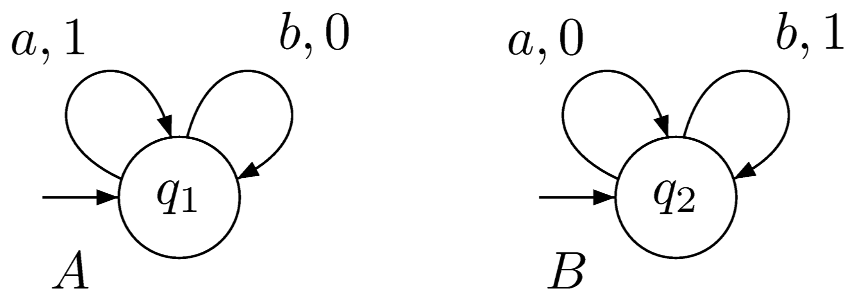
Assume that $L_m = \min(L_a, L_b)$ is definable by a LimAvg automaton C.

Then, some a-cycle or b-cycle in C has average weight >0.

(consider the word $(a^m b^m)^\omega$ for m large)

Closure properties

There is no nondeterministic LimAvg automaton for the language $L_m = \min(L_a, L_b)$.



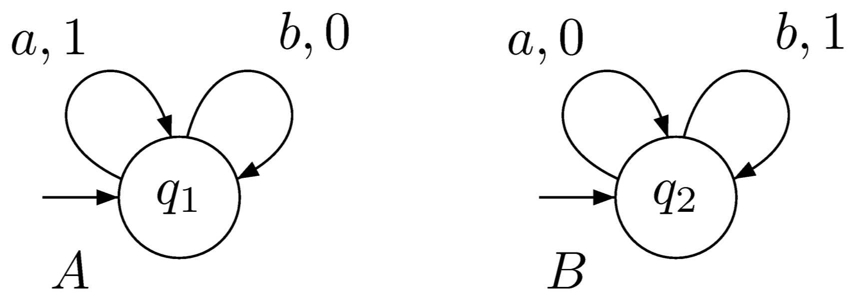
Assume that $L_m = \min(L_a, L_b)$ is definable by a LimAvg automaton C.

Then, some a-cycle (or b-cycle) in C has average weight >0.

Then, some word $w_C.a^\omega$ gets value >0…

Closure properties

There is no nondeterministic LimAvg automaton for the language $L_m = \min(L_a, L_b)$.



There is no nondeterministic Discounted automaton for the language $L_m = \min(L_a, L_b)$.

Proof: analogous argument.

Closure properties

	$\cup$	$\cap$	$\Sigma^\omega \setminus L$	
	max	min	comp.	sum
$\substack{\text{D}\\\text{N}}$ Sup	$\surd$	$\surd$	$\times$	$\surd$
$\substack{\text{D}\\\text{N}}$ LimInf	$\surd$	$\surd$	$\times$	$\surd$
DLimSup	$\surd$	$\surd$	$\times$	$\surd$
NLimSup	$\surd$	$\surd$	$\surd$	$\surd$
DLimAvg	$\times$	$\times$		
NLimAvg	$\surd$	$\times$		
DDisc	$\times$	$\times$		
NDisc	$\surd$	$\times$		

Closure properties

	$\cup$ max	$\cap$ min	$\Sigma^\omega \setminus L$ comp.	sum
$\substack{\mathrm{D}\\\mathrm{N}}$ Sup	$\surd$	$\surd$	$\times$	$\surd$
$\substack{\mathrm{D}\\\mathrm{N}}$ LimInf	$\surd$	$\surd$	$\times$	$\surd$
DLimSup	$\surd$	$\surd$	$\times$	$\surd$
NLimSup	$\surd$	$\surd$	$\surd$	$\surd$
DLimAvg	$\times$	$\times$		
NLimAvg	$\surd$	$\times$	$\times$	
DDisc	$\times$	$\times$		
NDisc	$\surd$	$\times$	$\times$	

$\min(L_1, L_2) = 1 - \max(1 - L_1, 1 - L_2)$

Closure properties

	$\cup$ max	$\cap$ min	$\Sigma^\omega \setminus L$ comp.	sum
$^{\text{D}}_{\text{N}}$ Sup	$\surd$	$\surd$	$\times$	$\surd$
$^{\text{D}}_{\text{N}}$ LimInf	$\surd$	$\surd$	$\times$	$\surd$
DLimSup	$\surd$	$\surd$	$\times$	$\surd$
NLimSup	$\surd$	$\surd$	$\surd$	$\surd$
DLimAvg	$\times$	$\times$	$\times$	$\times$
NLimAvg	$\surd$	$\times$	$\times$	$\times$
DDisc	$\times$	$\times$	$\surd$	$\surd$
NDisc	$\surd$	$\times$	$\times$	$\surd$

By analogous arguments (analysis of cycles).

Conclusion

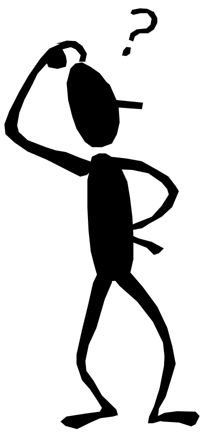
- Quantitative generalization of languages to model programs/systems more accurately.
- LimAvg and Disc$_\lambda$: deciding language inclusion is open;
- Simulation is a decidable over-approximation.
- Expressive power classification:
    - DBW and LimAvg are incomparable;
    - LimAvg and Disc cannot be determinized.
- Closure properties.

Other lines of work

- Cut-point languages, stability/robustness (LICS'09)
- Alternating/Probabilistic extensions (CONCUR'09, FCT'09)
- Robust class of Limit-Average automata (ongoing work)
- Open problem: quantitative universality (partial results)
- Other/equivalent formalisms for quantitative specification ?

The end

Thank you !



Questions ?