Quantitative Languages

Krishnendu Chatterjee, UCSC

Laurent Doyen, EPFL

Tom Henzinger, EPFL

CSL 2008

Languages

A language

$$L(A) \subseteq \Sigma^{\omega}$$

can be viewed as a boolean function:

$$L_A: \Sigma^{\omega} \rightarrow \{0,1\}$$

Model-Checking

Model-checking problem

Input: Model A of the program

Model B of the specification

Question: does the program A satisfy the specification B ?

$$A \overset{?}{\models} B$$

Automata & Languages

Model-checking as language inclusion

Input: finite automata A and B

Question: is $L(A) \subseteq L(B)$?

$A \overset{?}{\models} B$

Automata & Languages

Model-checking as language inclusion

Input: finite automata A and B

Question: is $L_A(w) \leq L_B(w)$ for all words w ?

Languages are boolean

Quantitative languages

A quantitative language (over infinite words) is a function

$$L : \Sigma^{\omega} \longrightarrow \mathbb{R}$$

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),
- a probability, etc.

Quantitative languages

Quantitative language inclusion

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

Example:

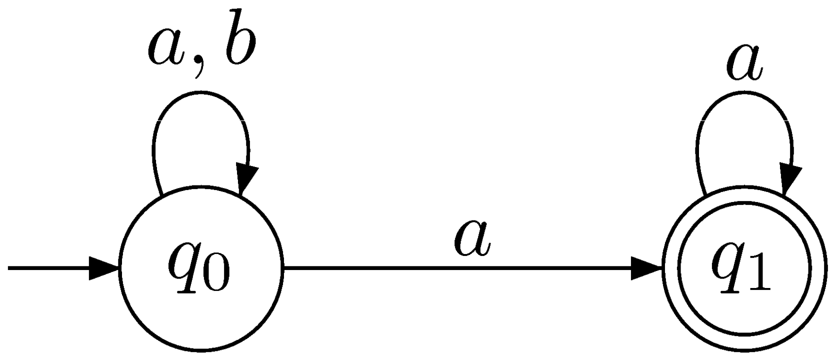
$L_A(w)$	$L_B(w)$
peak resource consumption	resource bound
Long-run average response time	Average response-time requirement

Outline

- Motivation
- Weighted automata
- Decision problems
- Expressive power

Automata

Boolean languages are generated by finite automata.

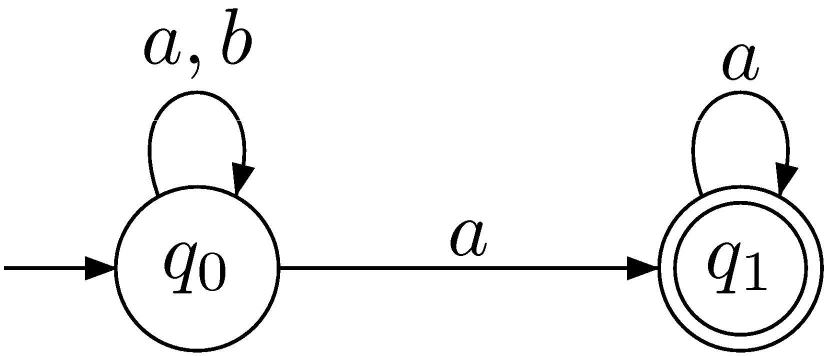


$\mathsf{L} = (a+b)^* \cdot a^\omega$

Nondeterministic Büchi automaton

Automata

Boolean languages are generated by finite automata.



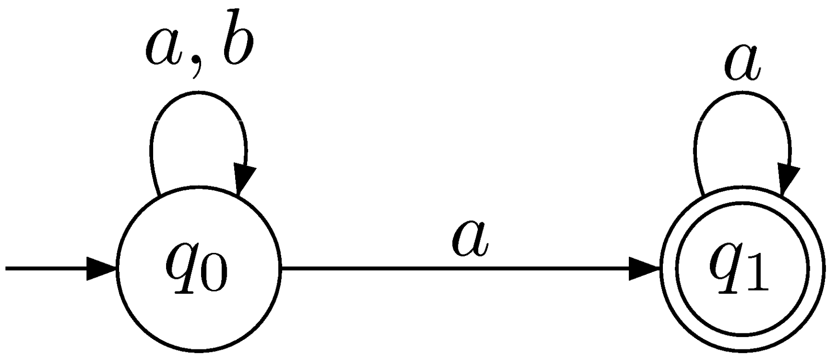
$L = (a+b)^* \cdot a^\omega$

Nondeterministic Büchi automaton

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

Automata

Boolean languages are generated by finite automata.



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

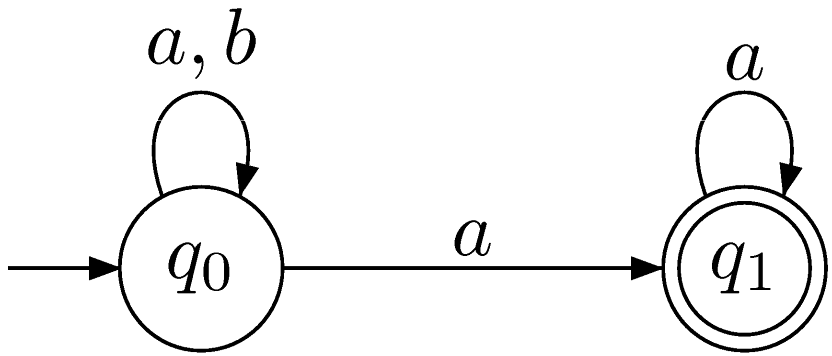
Nondeterministic Büchi automaton

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.



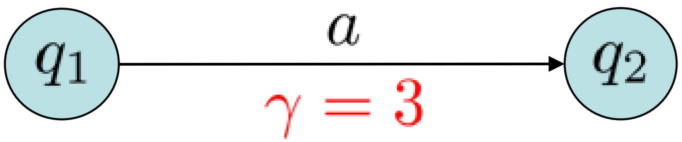
$\mathsf{L} = (a+b)^* \cdot a^\omega$

Nondeterministic Büchi automaton

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

Weighted automata

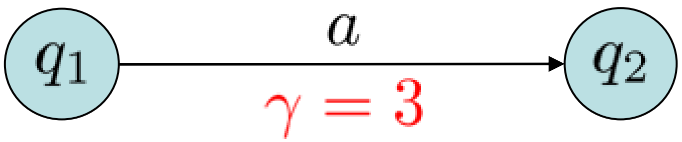
Quantitative languages are generated by weighted automata.



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

Weighted automata

Quantitative languages are generated by weighted automata.



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 $\mathsf{Val} : \mathbb{Q}^\omega \longrightarrow \mathbb{R}$ is a value function

Some value functions

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

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

Some value functions

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

- $\mathsf{Sup}(v) = \sup\{v_n \mid n \geq 0\}$;
- $\mathsf{LimSup}(v) = \limsup_{n \to \infty} v_n = \lim_{n \to \infty} \sup\{v_i \mid i \geq n\}$;
- $\mathsf{LimInf}(v) = \liminf_{n \to \infty} v_n = \lim_{n \to \infty} \inf\{v_i \mid i \geq n\}$;
- $\mathsf{LimAvg}(v) = \liminf_{n \to \infty} \frac{1}{n} \cdot \sum_{i=0}^{n-1} v_i$;
- given a discount factor $0 < \lambda < 1$, $\mathsf{Disc}_\lambda(v) = \sum_{i=0}^{\infty} \lambda^i \cdot v_i$.

Outline

- Motivation
- Weighted automata
- Decision problems
- Expressive power

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 ?

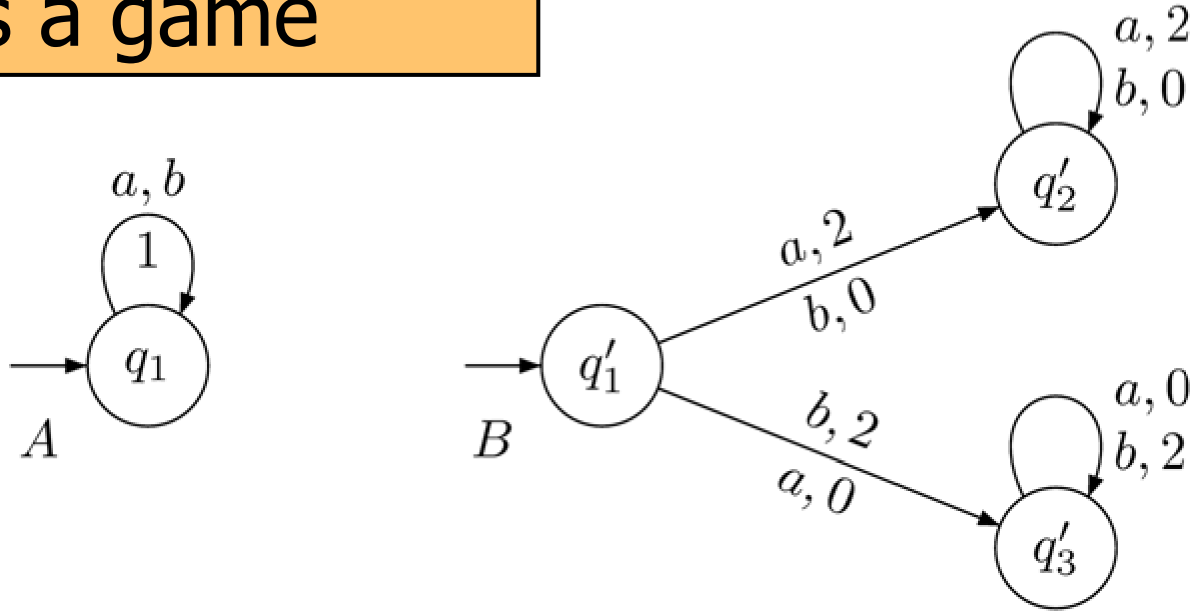
- PSPACE-complete for Sup, LimSup and LimInf.

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

- open question for nondeterministic automata.

Language-inclusion game

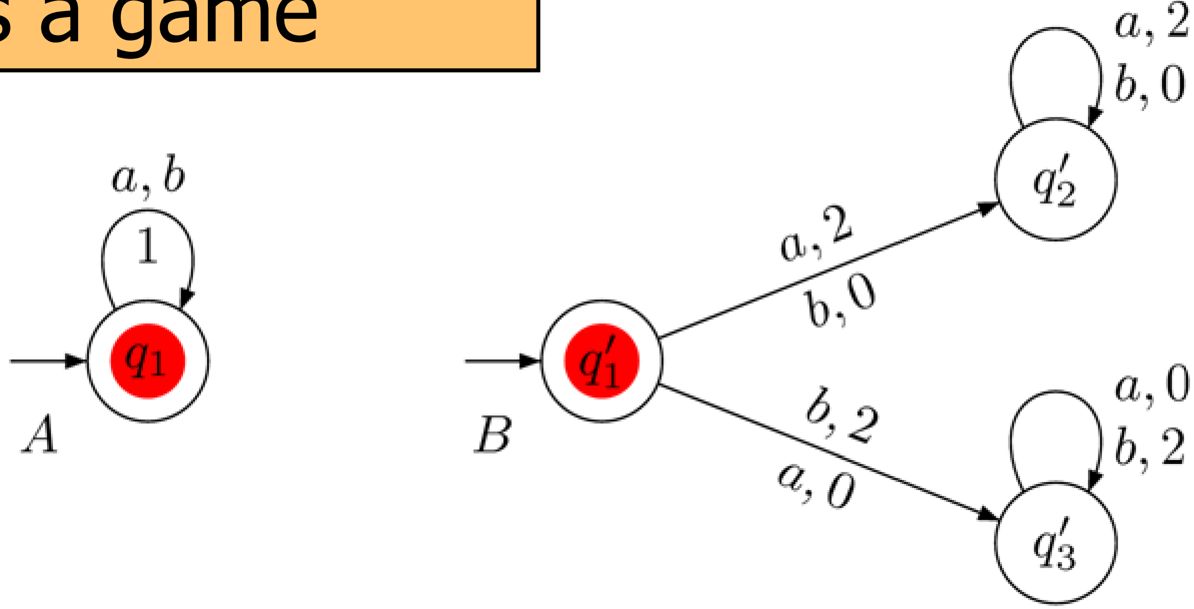
Language inclusion as a game



Discounted-sum automata, λ=3/4

Language-inclusion game

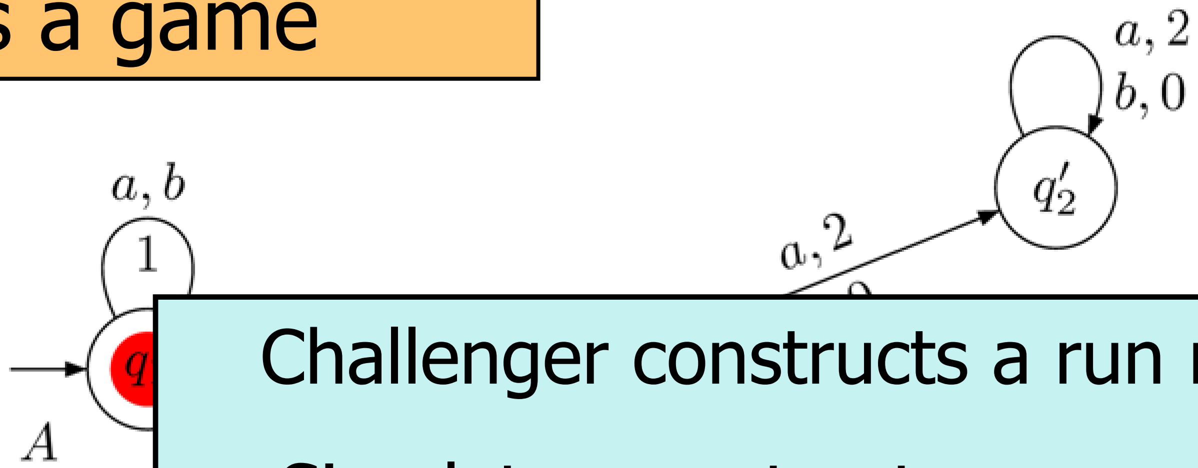
Language inclusion as a game



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.

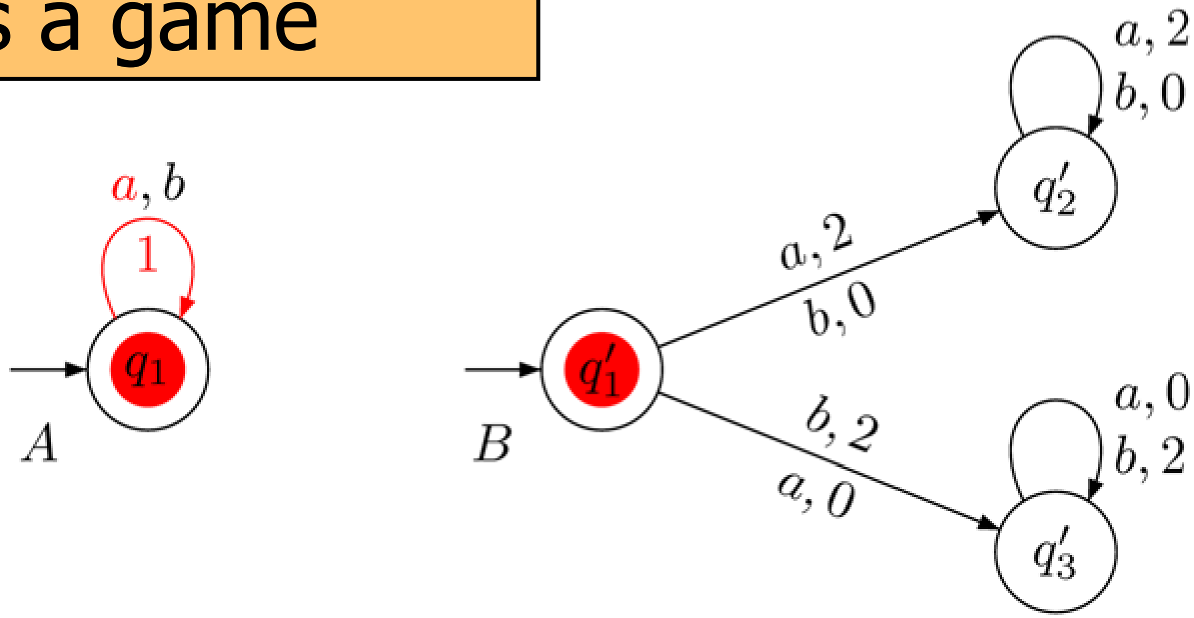
Challenger wins if $Val(r_1) > Val(r_2)$.

Challenger: q_1

Simulator: q'_1

Language-inclusion game

Language inclusion as a game

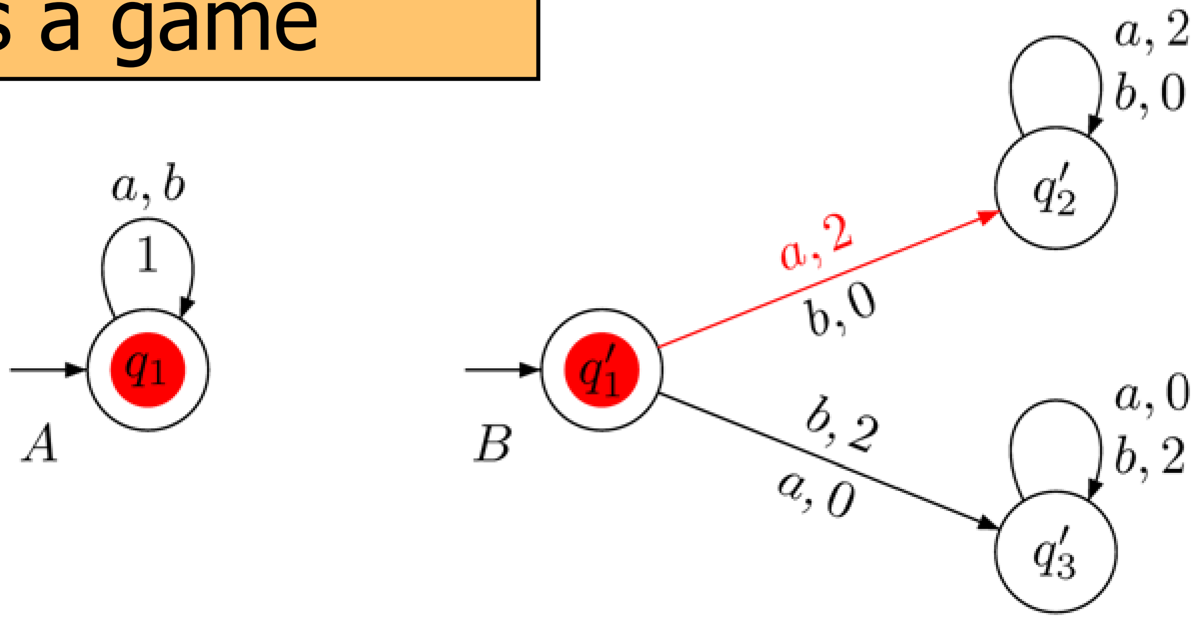


Challenger: $q_1 \xrightarrow[1]{a} q_1$

Simulator: q'_1

Language-inclusion game

Language inclusion as a game

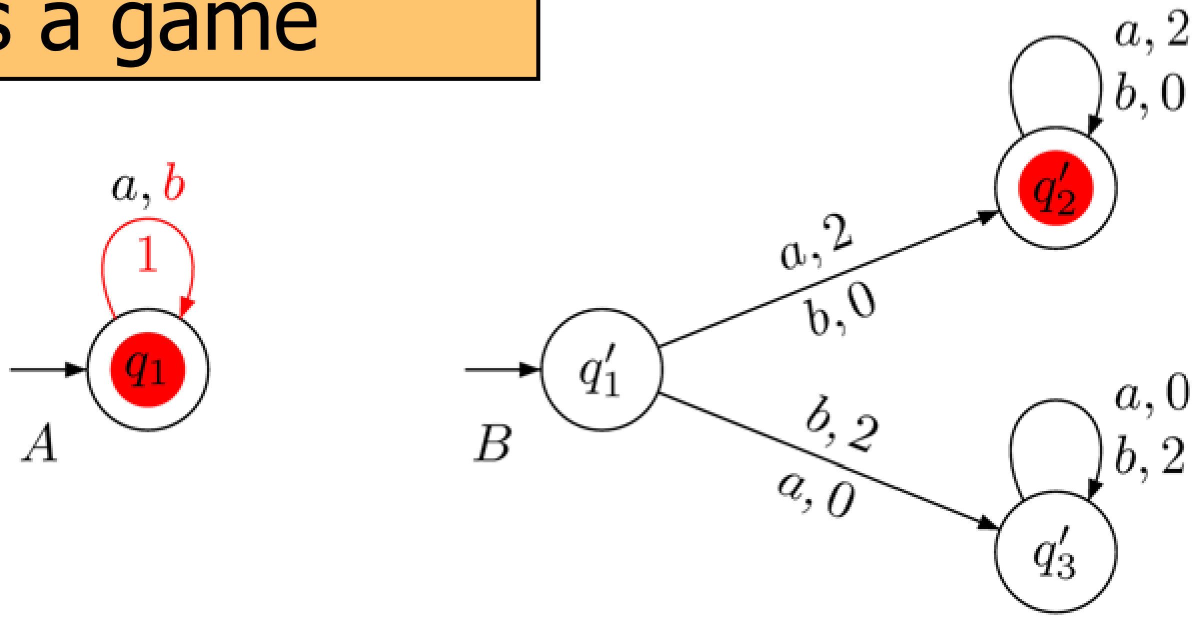


Challenger: $q_1 \xrightarrow[1]{a} q_1$

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

Language-inclusion game

Language inclusion as a game

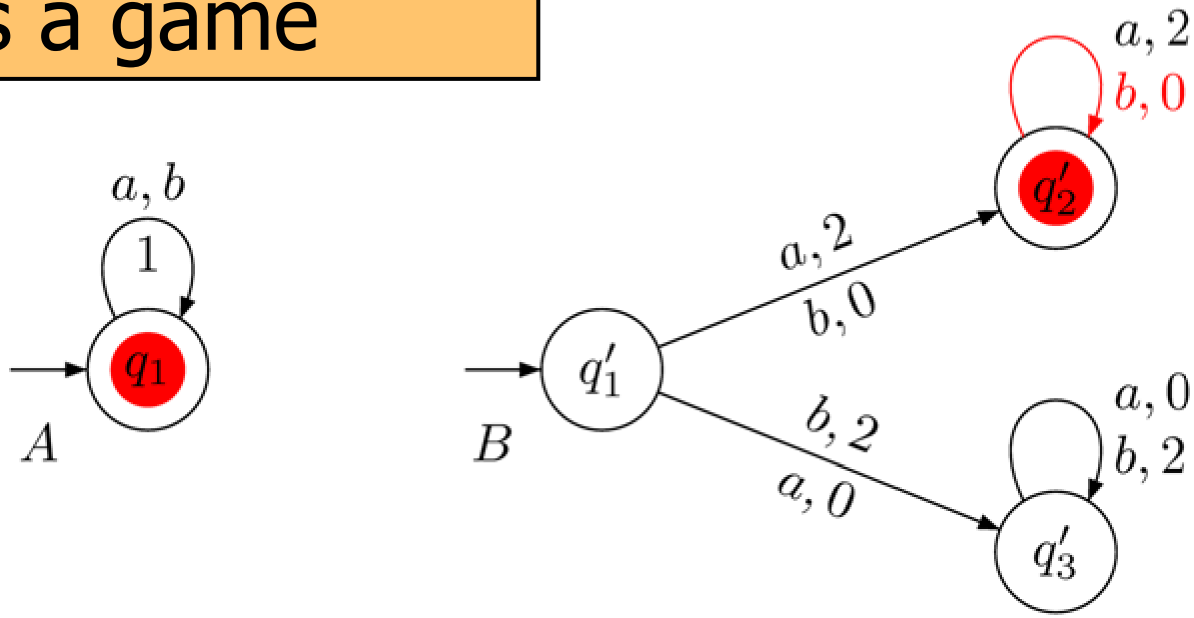


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1$

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

Language-inclusion game

Language inclusion as a game

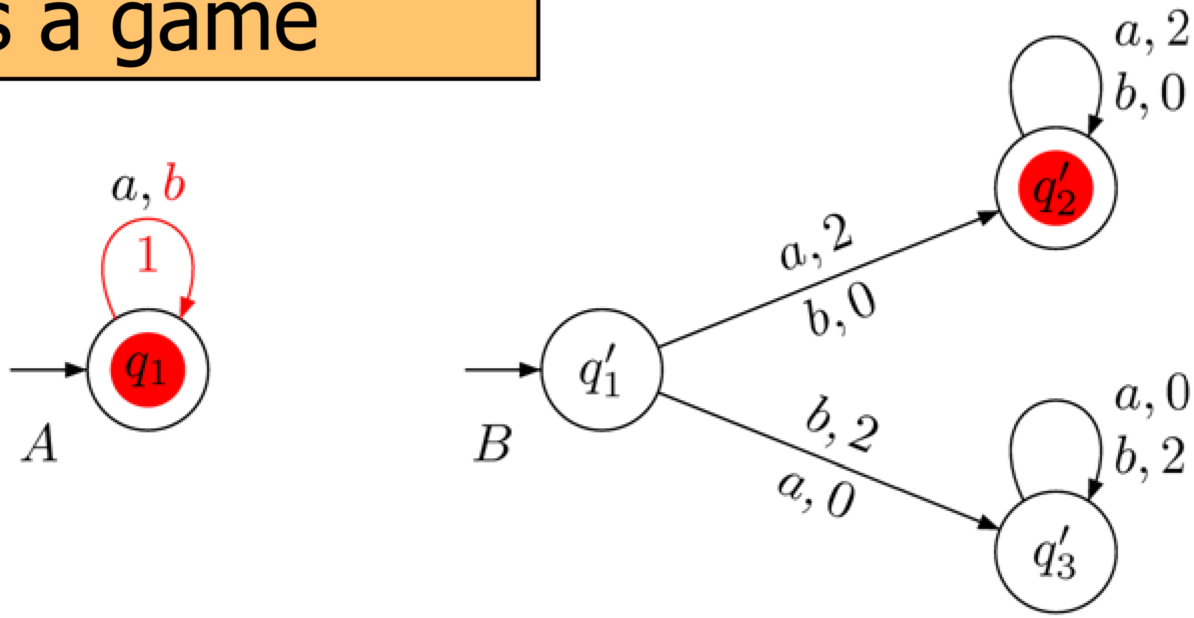


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1$

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

Language-inclusion game

Language inclusion as a game

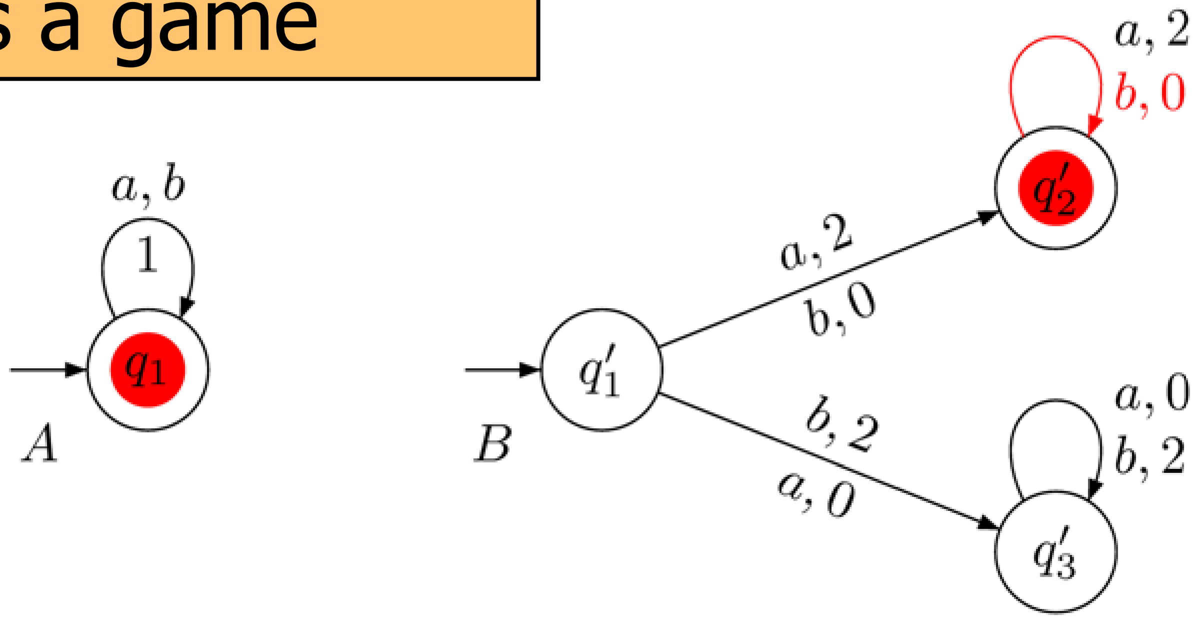


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1 \xrightarrow[1]{b} \cdots$

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

Language-inclusion game

Language inclusion as a game

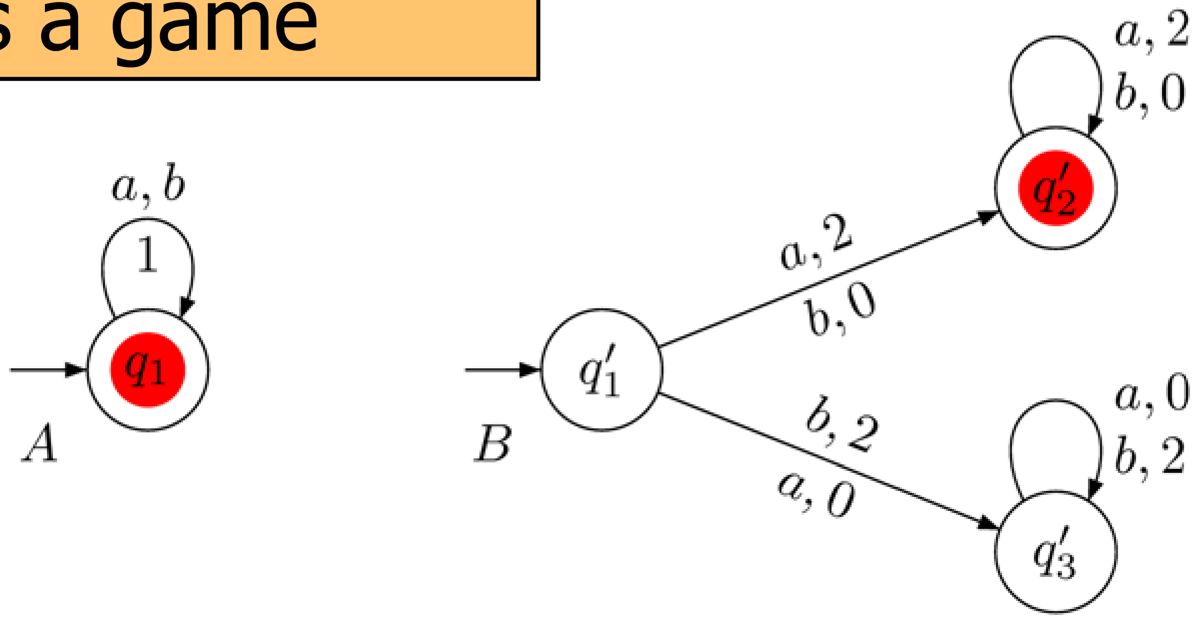


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1 \xrightarrow[1]{b} \cdots$

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

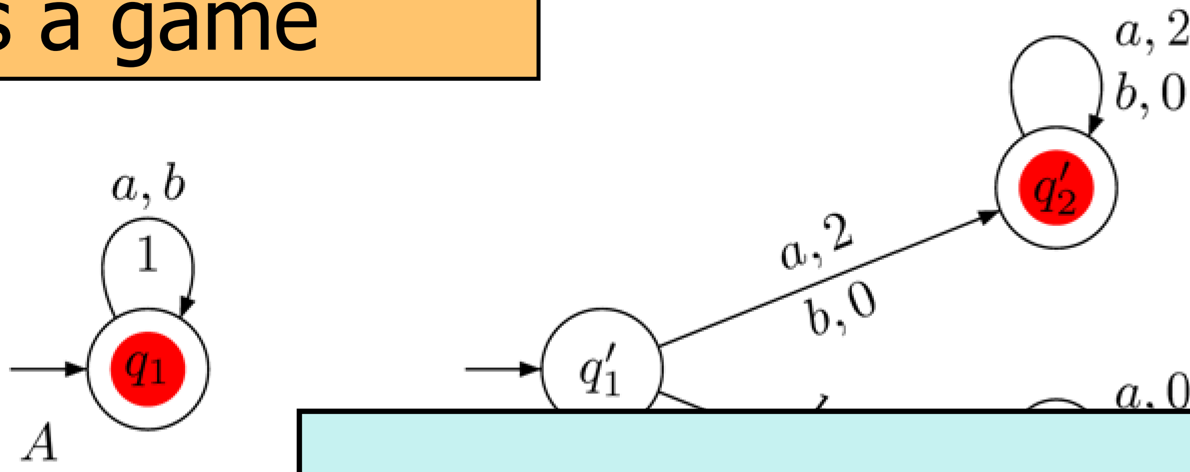


Challenger: $q_1 \xrightarrow[1]{a} q_1 \xrightarrow[1]{b} q_1 \xrightarrow[1]{b} \cdots$ $\quad$ $\text{Disc}_{\frac{3}{4}}(1, 1, 1, \ldots) = \frac{1}{1-\frac{3}{4}} = 4.$

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

Language-inclusion game

Language inclusion as a game



Challenger wins since 4>2.

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

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

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

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

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

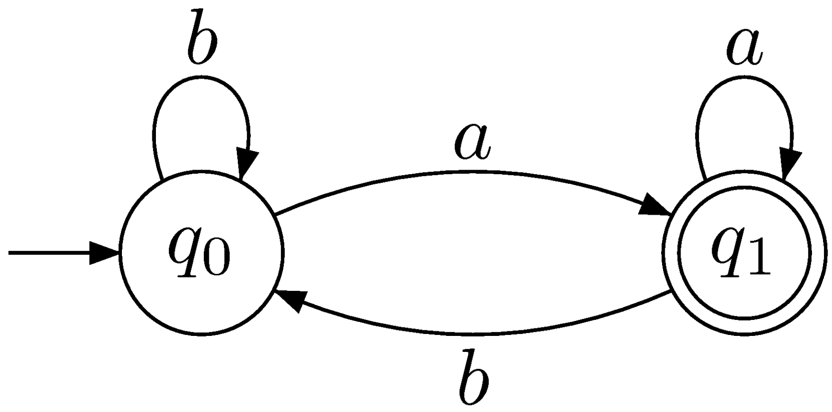
For discounted-sum, **prefixes** provide good approximations of the value.

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



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

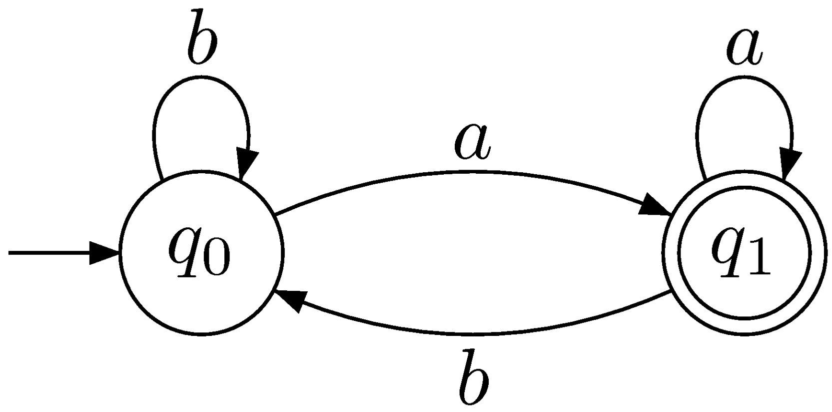
"infinitely many a"

Deterministic Büchi automaton

Assume that L is definable by a LimAvg automaton A.

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

Büchi does not reduce to LimAvg



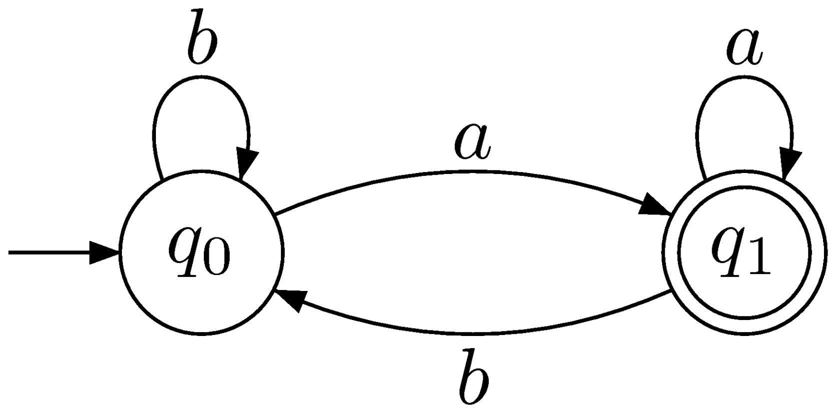
$\mathsf{L}_1 = (\Sigma^* \cdot a)^\omega$

"infinitely many a"

Deterministic Büchi automaton

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



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

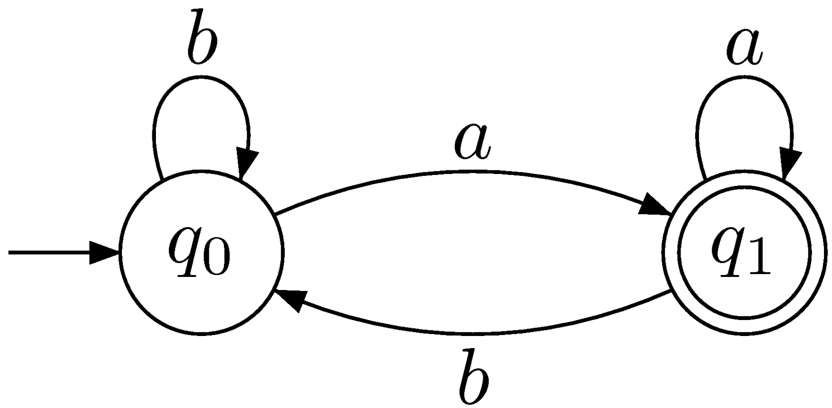
"infinitely many a"

Deterministic Büchi automaton

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



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

"infinitely many a"

Deterministic Büchi automaton

Let $w_n = (a \cdot b^n)^\omega$ We have $L_1(w_n) = 1$

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

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

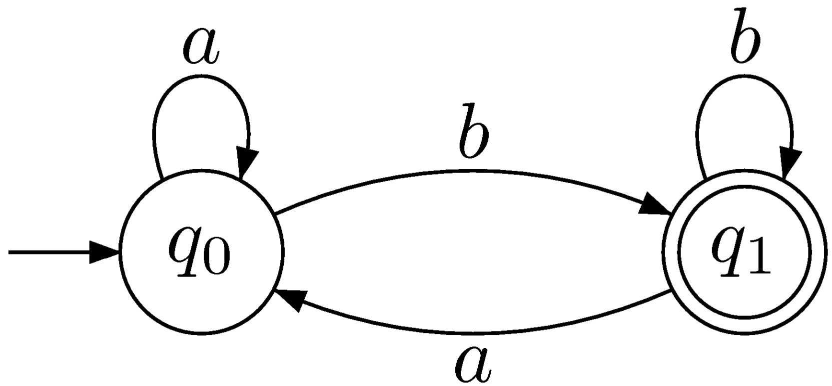
and A cannot exist !

(co)Büchi and LimAvg

det. Büchi cannot be reduced to LimAvg.

By analogous arguments,

det. coBüchi cannot be reduced to det. LimAvg.

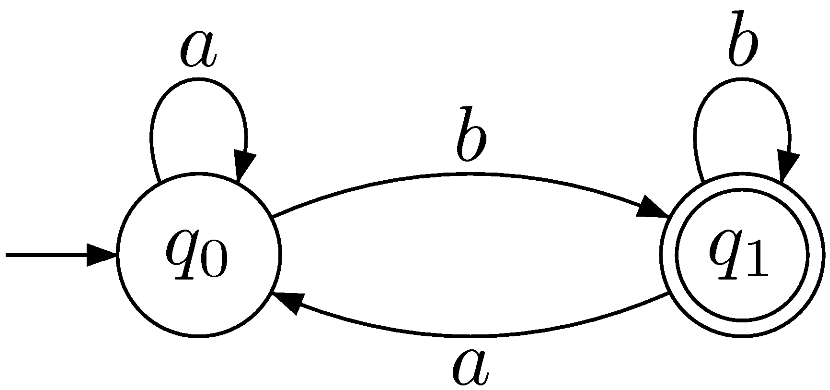


$\mathsf{L}_2 = \Sigma^* \cdot b^\omega$

"finitely many a"

Deterministic coBüchi automaton

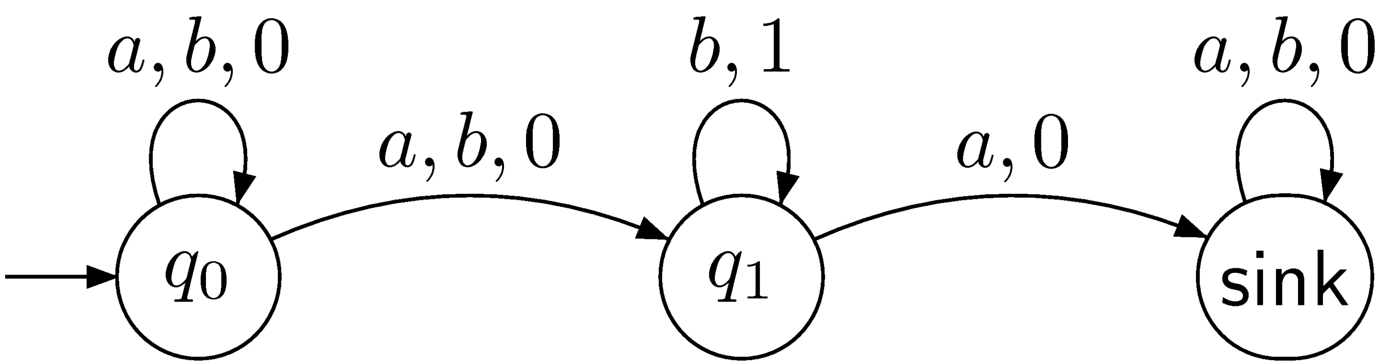
(co)Büchi and LimAvg



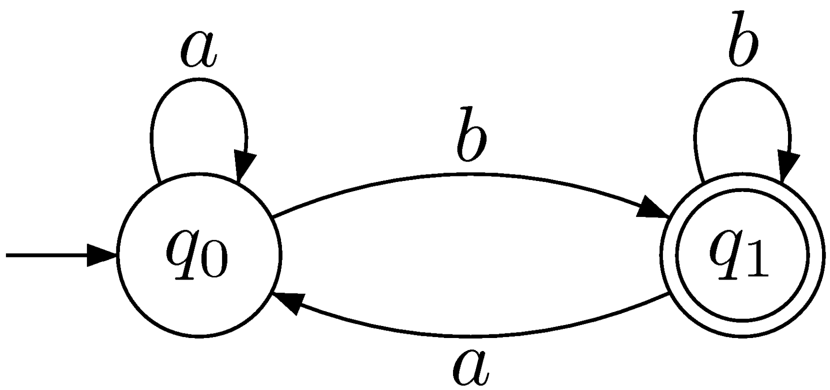
Det. coBüchi automaton

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

L_2 is defined by the following nondet. LimAvg automaton:



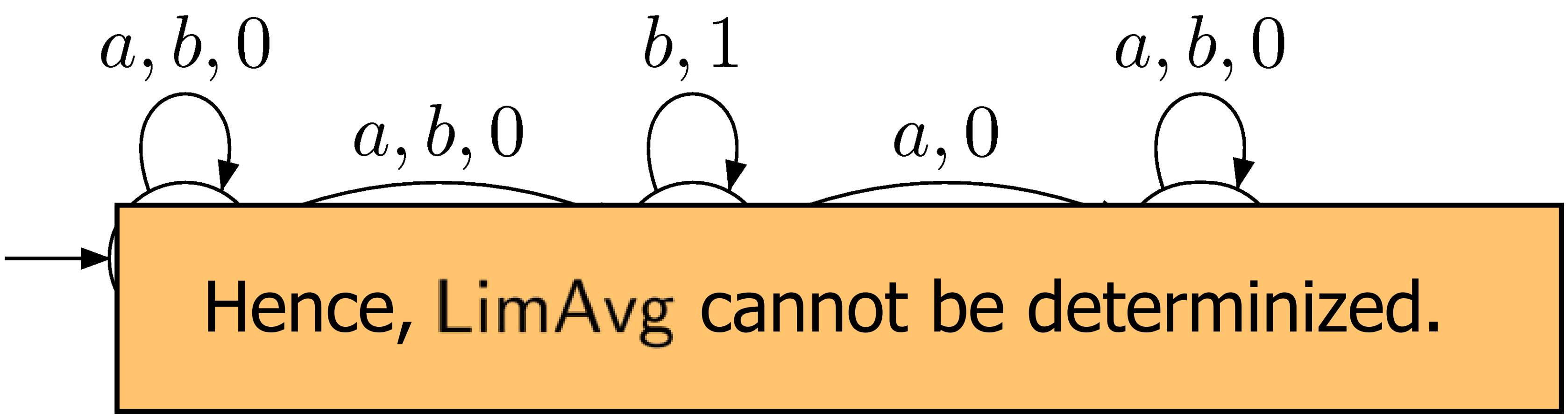
(co)Büchi and LimAvg



Det. coBüchi automaton

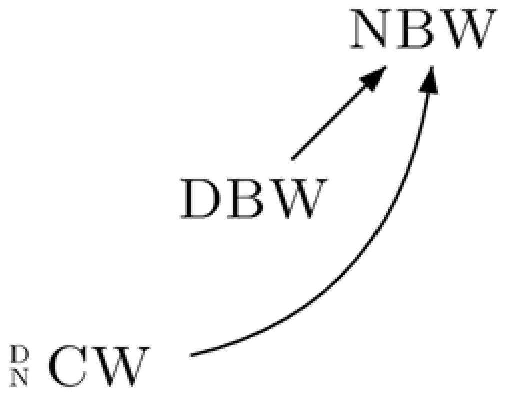
$L_2 = \Sigma^* \cdot b^\omega$

L_2 is defined by the following nondet. LimAvg automaton:

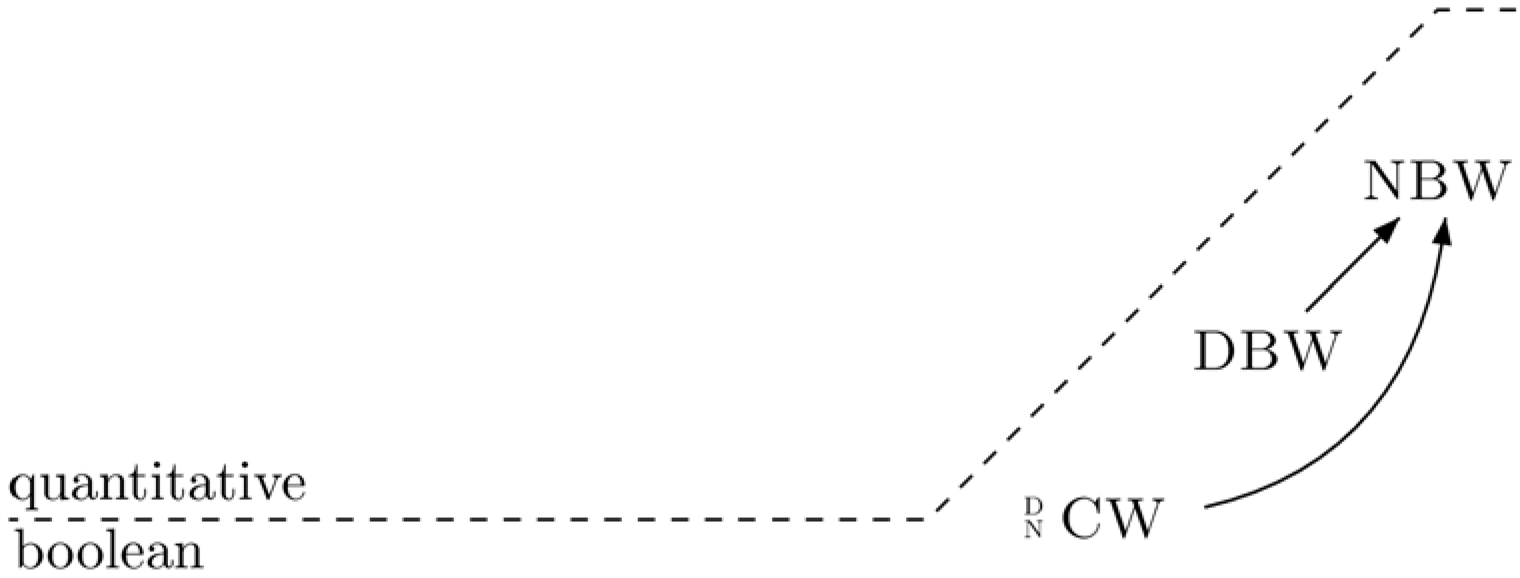


Hence, LimAvg cannot be determinized.

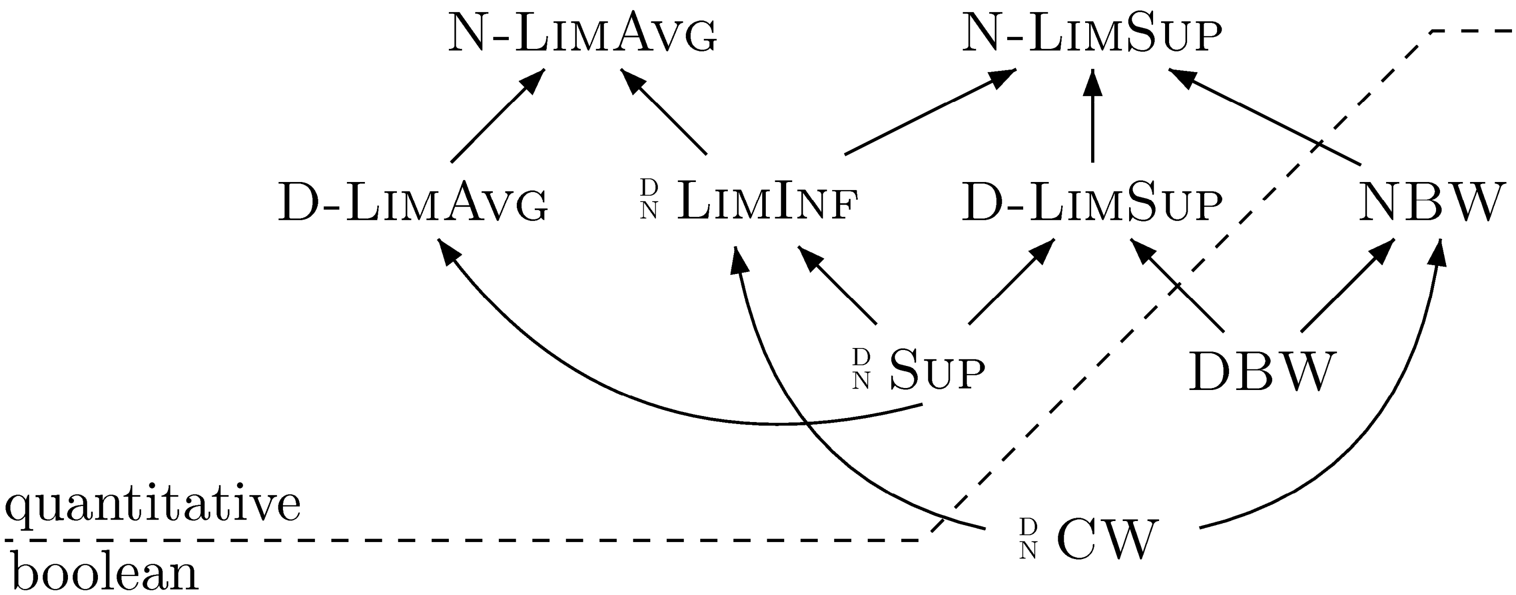
Reducibility relations



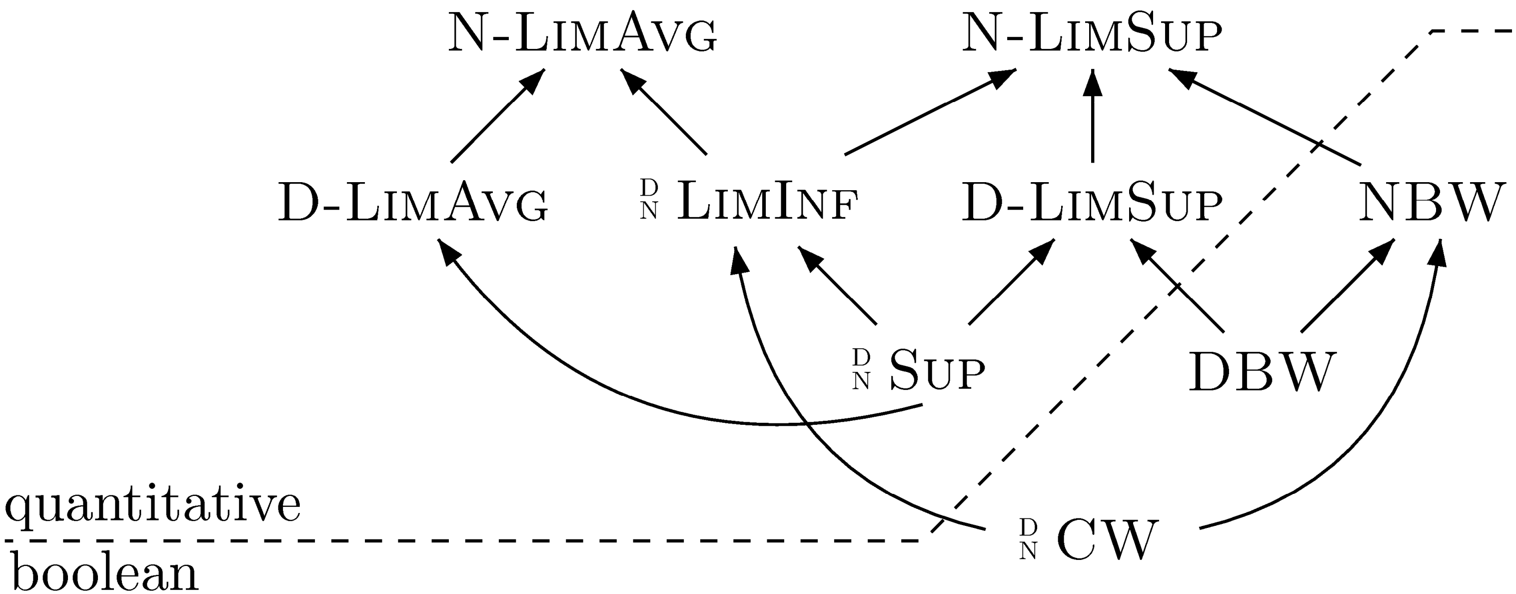
Reducibility relations



Reducibility relations



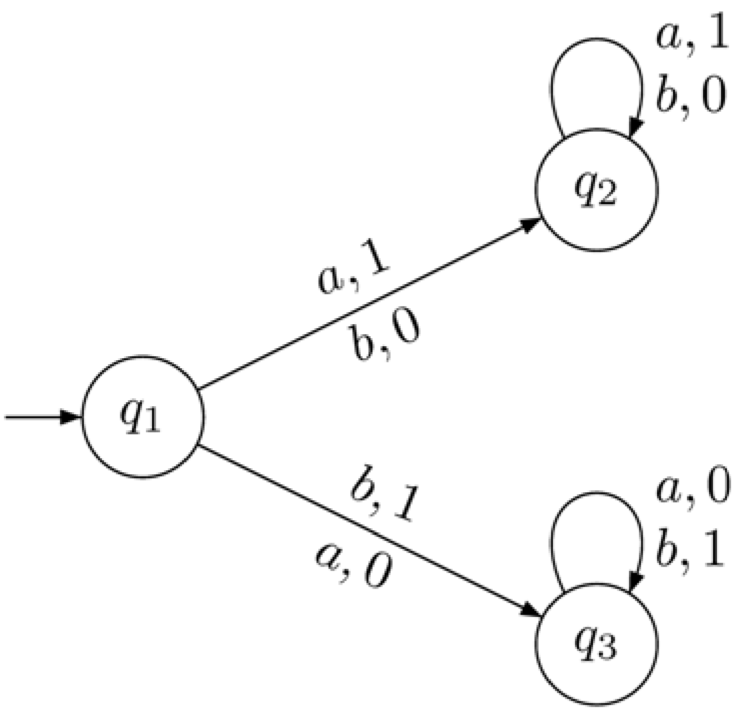
Reducibility relations



What about Discounted Sum ?

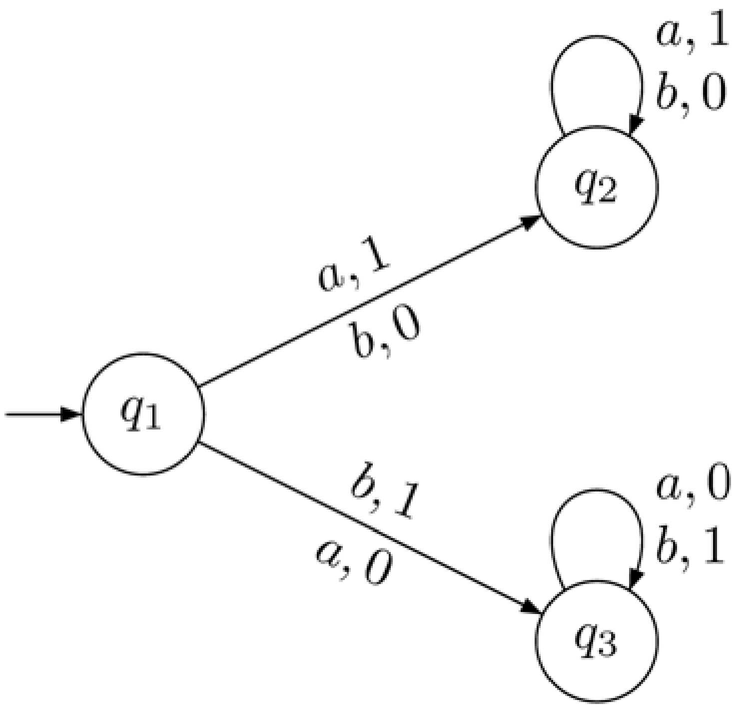
Last result

Disc_λ cannot be determinized.



λ=3/4

Disc$_\lambda$ cannot be determinized



Value of a word w :

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

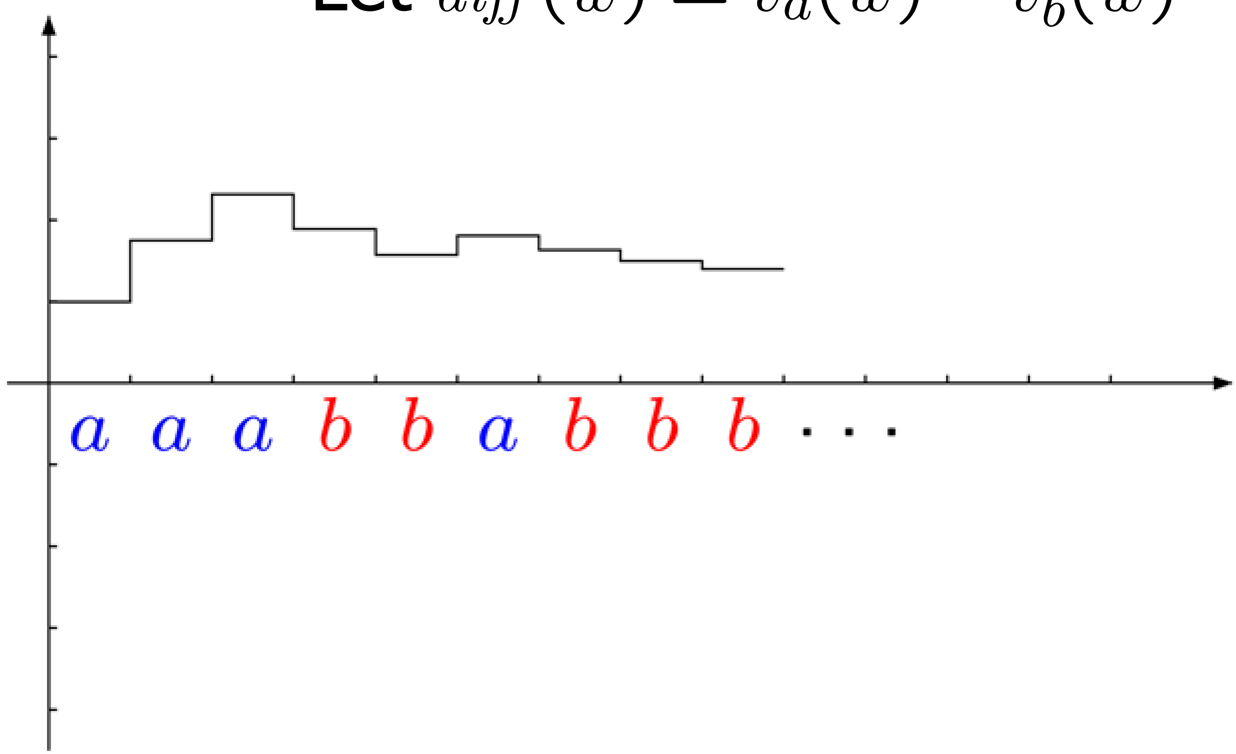
$$v_a(w) = \sum_{i | w_i = a}^{|w|} \lambda^i$$ disc. sum of a's

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

Disc$_\lambda$ cannot be determinized

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

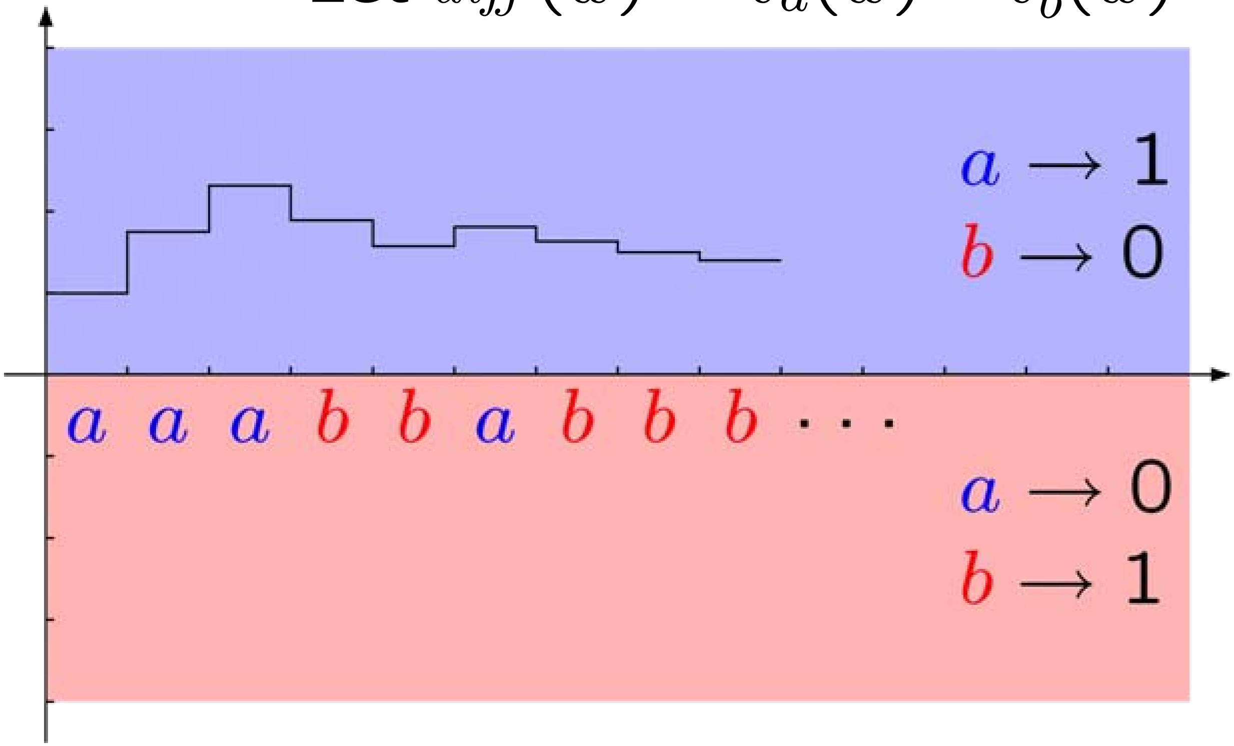
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

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

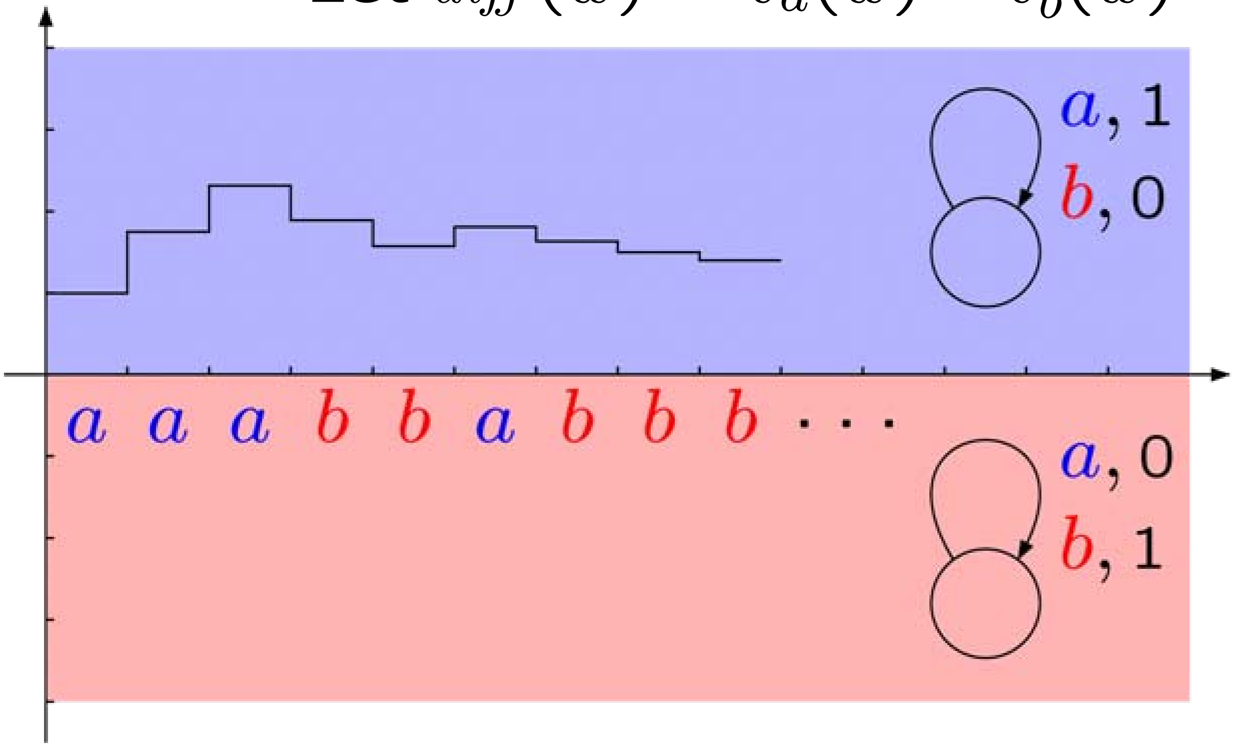
Let $diff(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

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

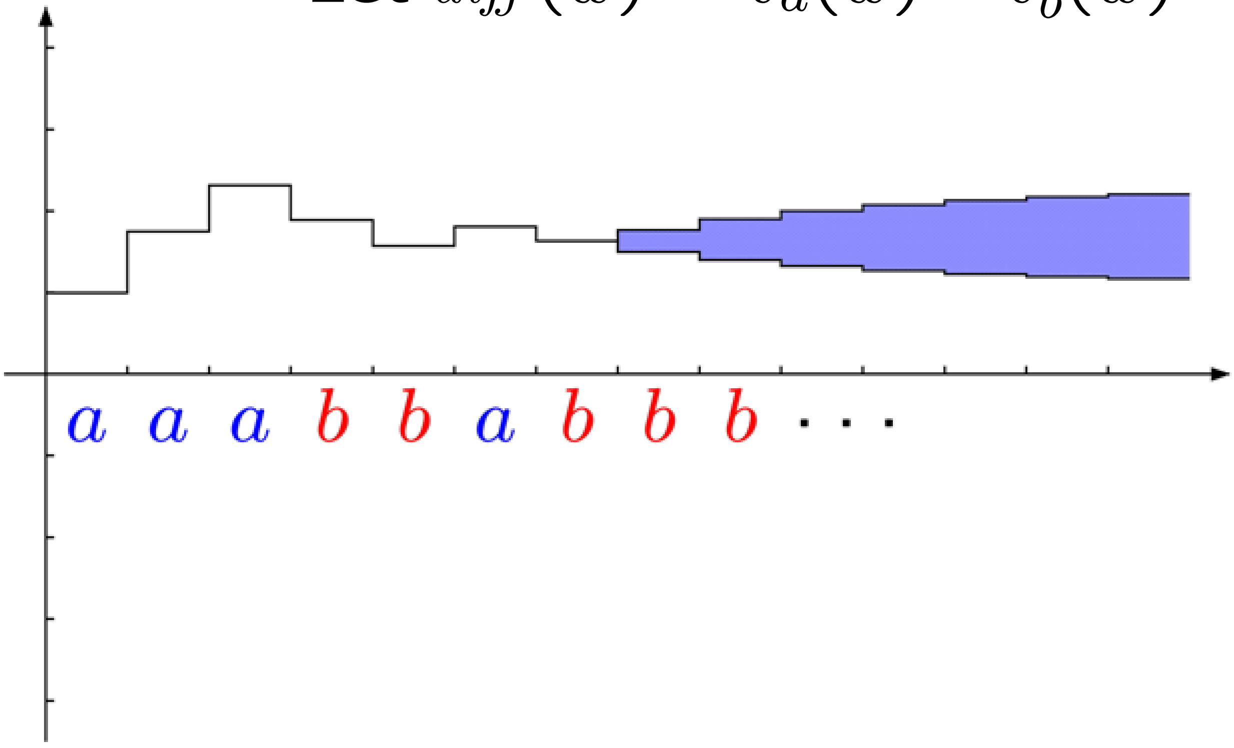
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc_λ cannot be determinized

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

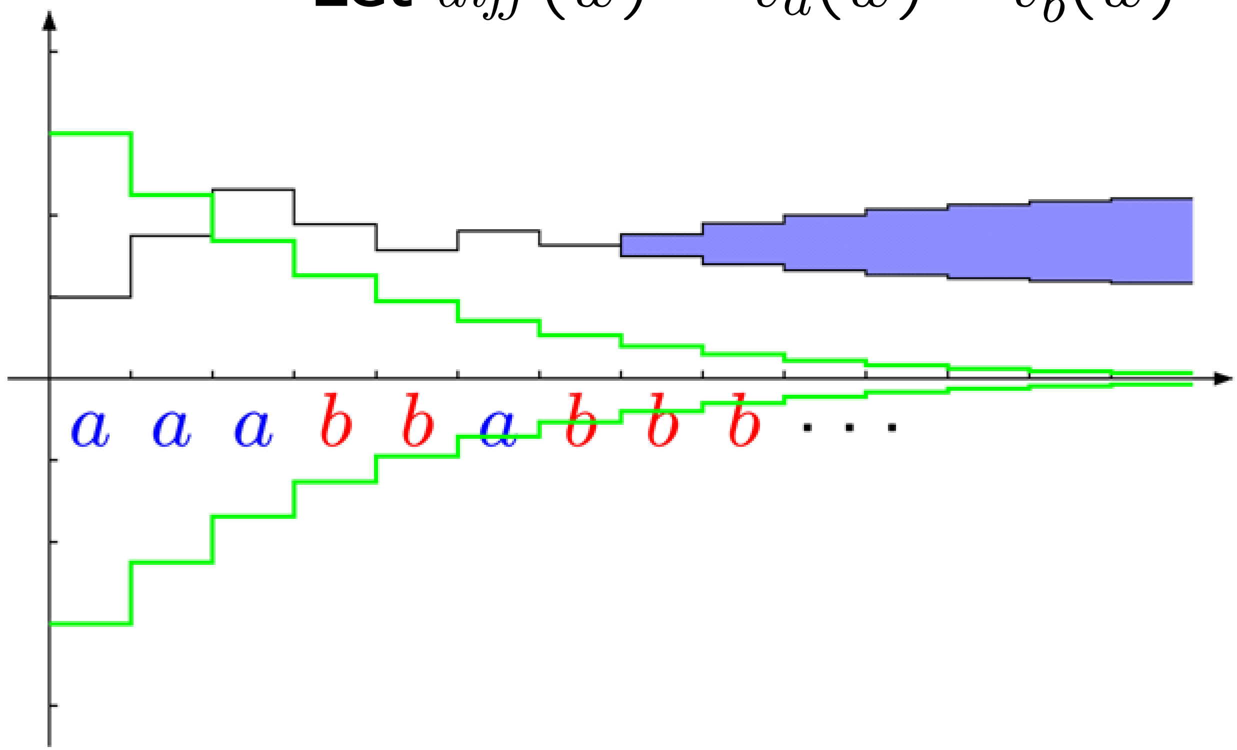
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

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

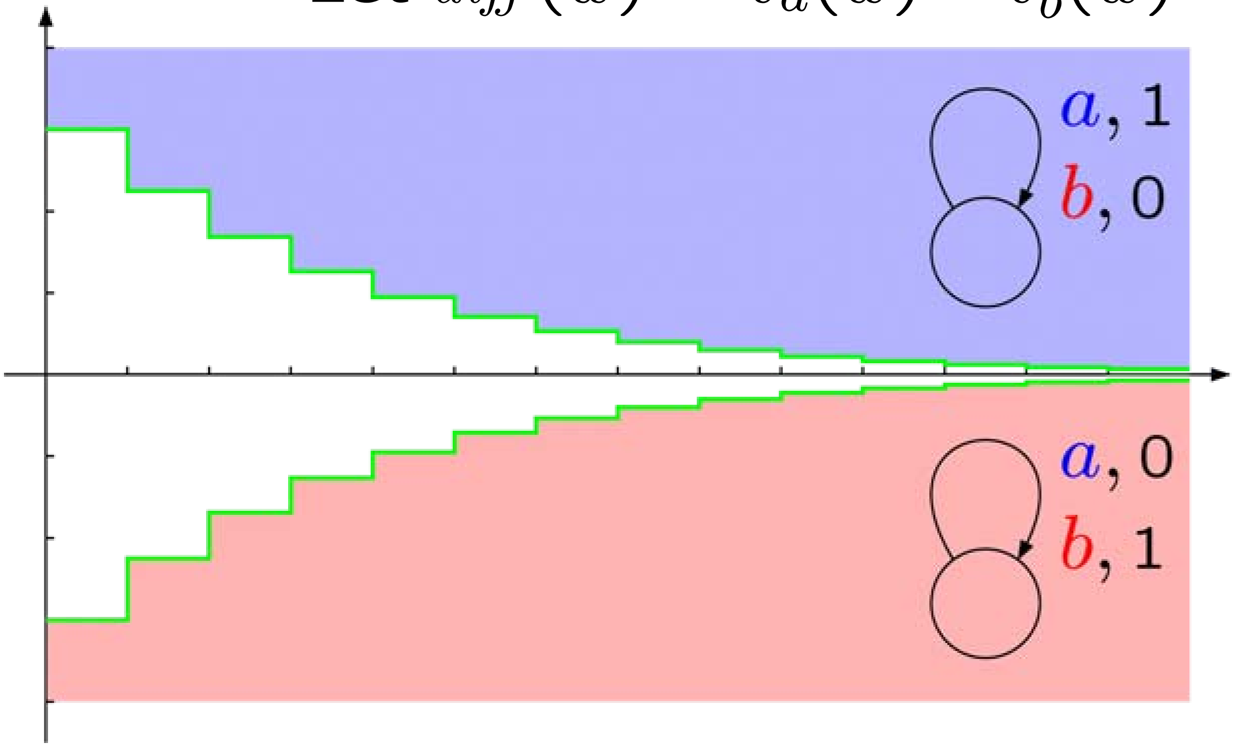
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

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

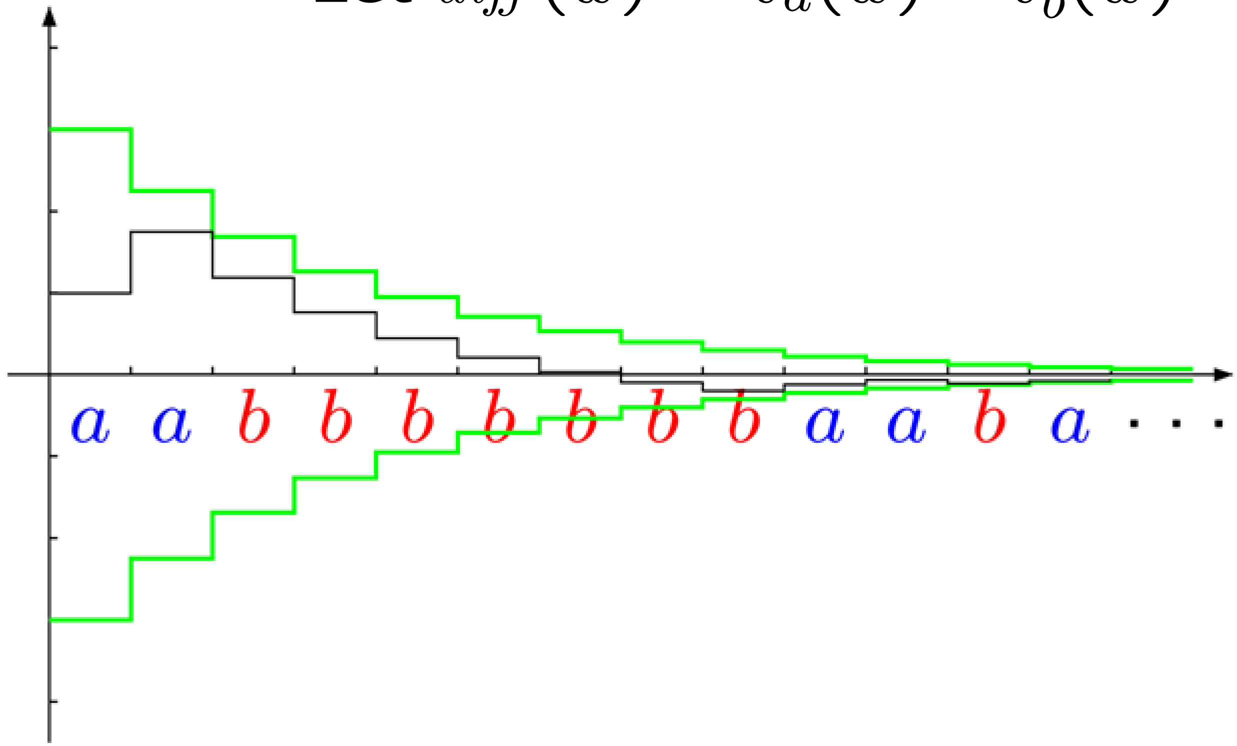
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

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

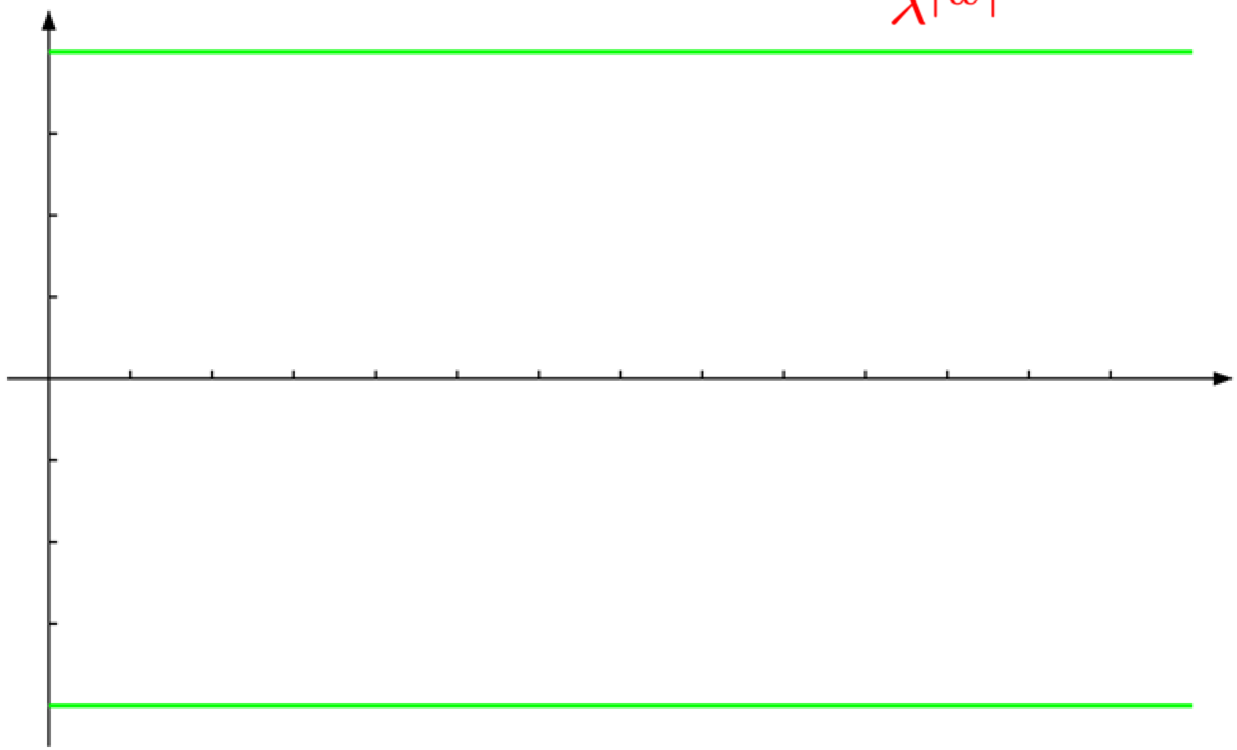
Let $\mathit{diff}(w) = v_a(w) - v_b(w)$



Disc$_\lambda$ cannot be determinized

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

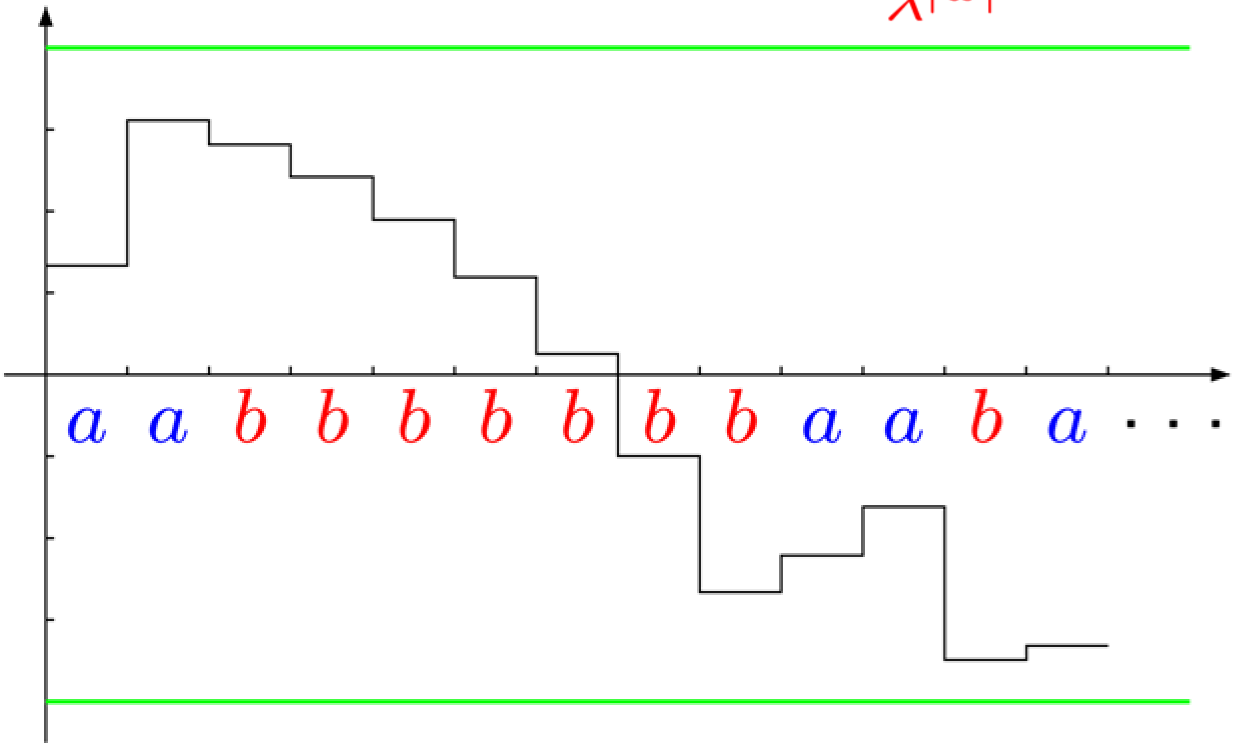
Let $diff(w) = \dfrac{v_a(w) - v_b(w)}{\lambda^{|w|}}$



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



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

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

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

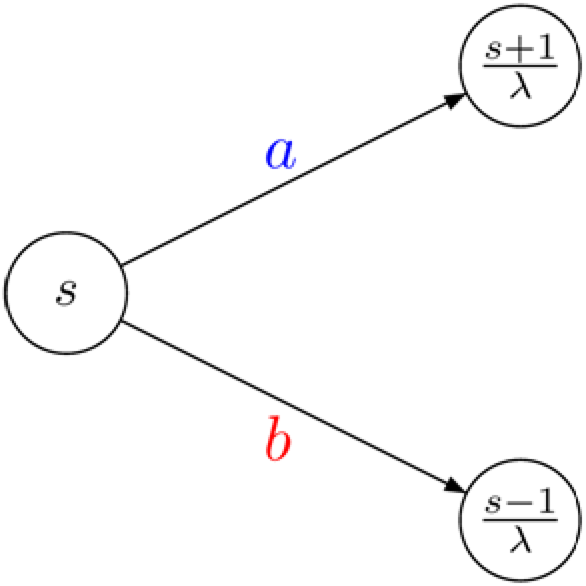
Disc$_\lambda$ cannot be determinized

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

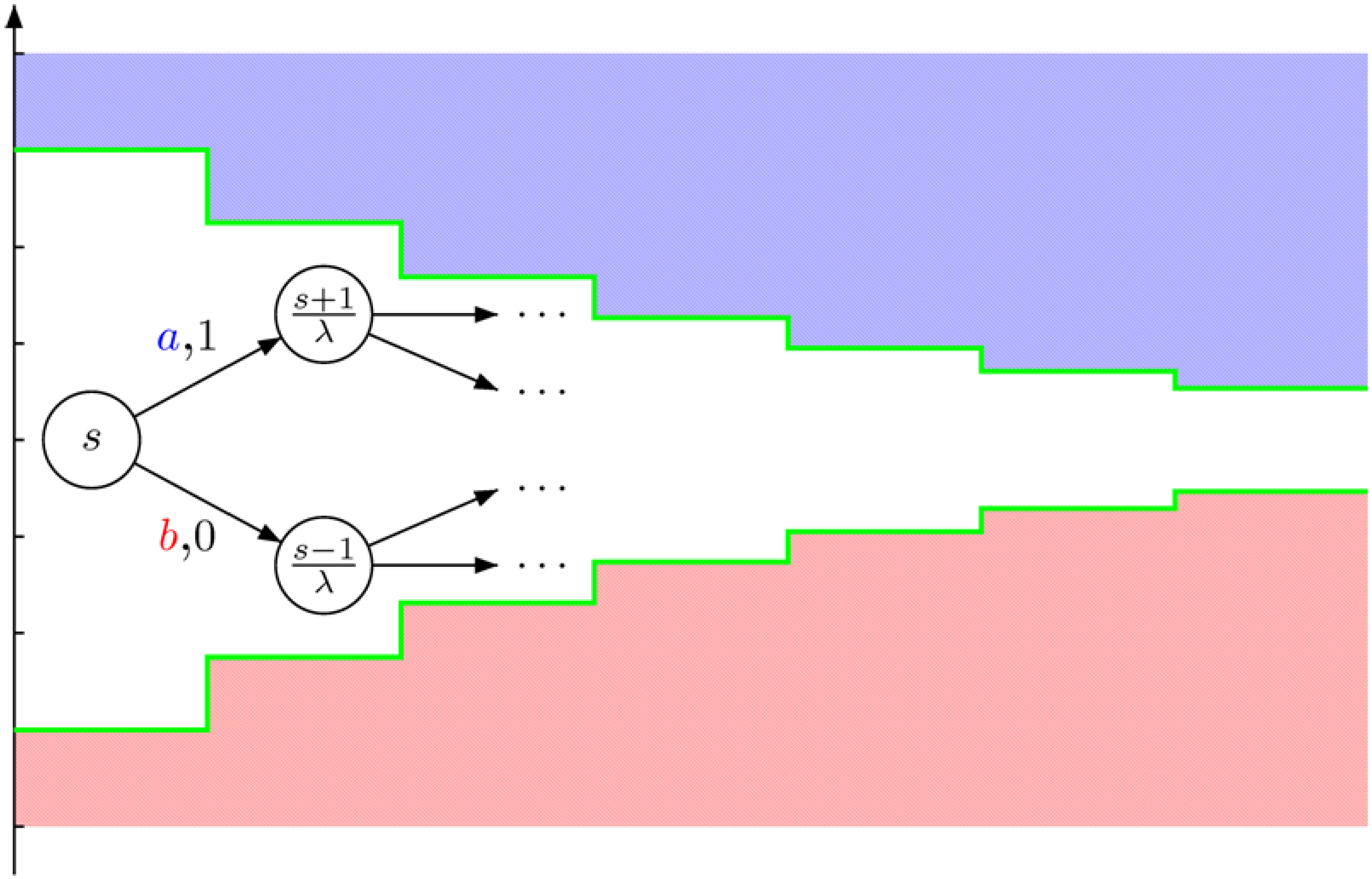
Let $\mathit{diff}(w) = \dfrac{v_a(w) - v_b(w)}{\lambda^{|w|}}$

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

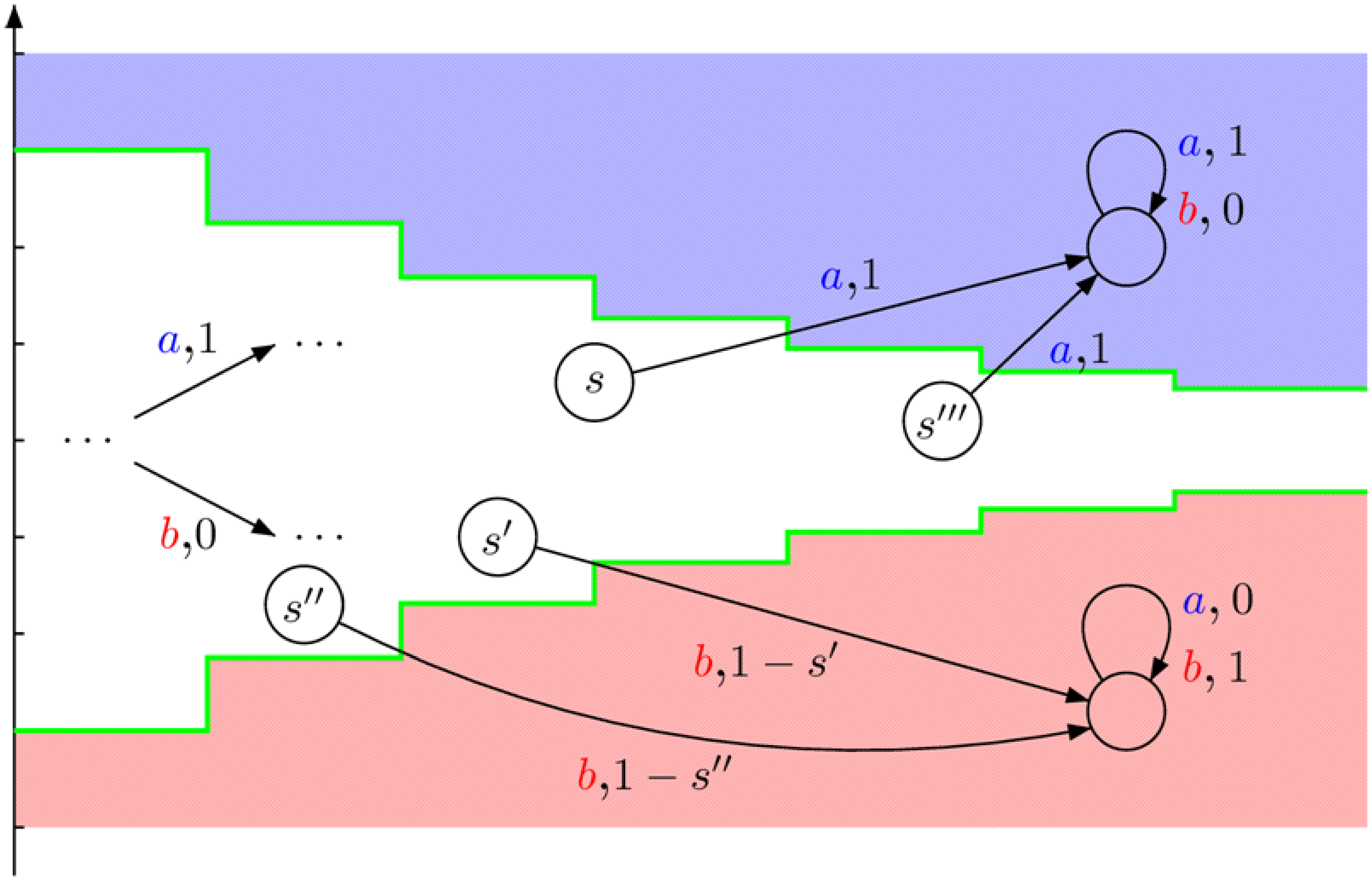
then $\begin{cases} \mathit{diff}(w \cdot a) & = & \frac{s+1}{\lambda} \\ \mathit{diff}(w \cdot b) & = & \frac{s-1}{\lambda} \end{cases}$



Disc$_\lambda$ cannot be determinized



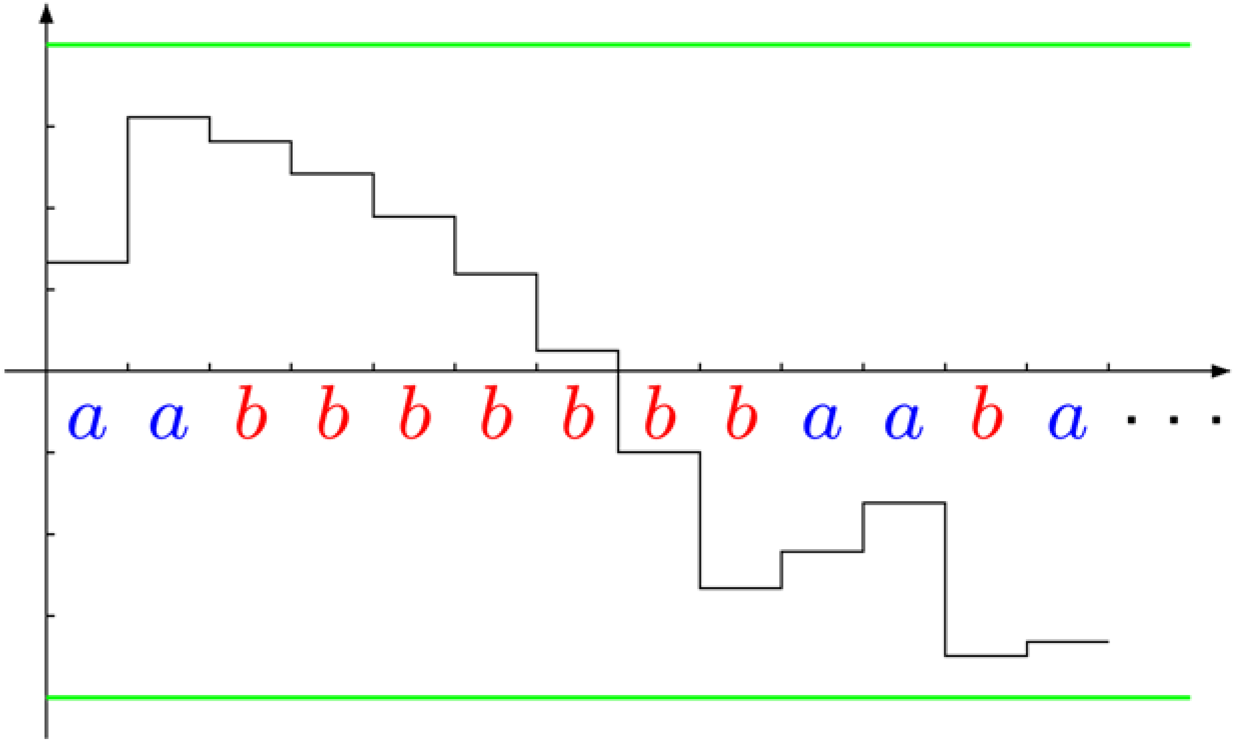
Disc_λ cannot be determinized



Disc$_\lambda$ cannot be determinized

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

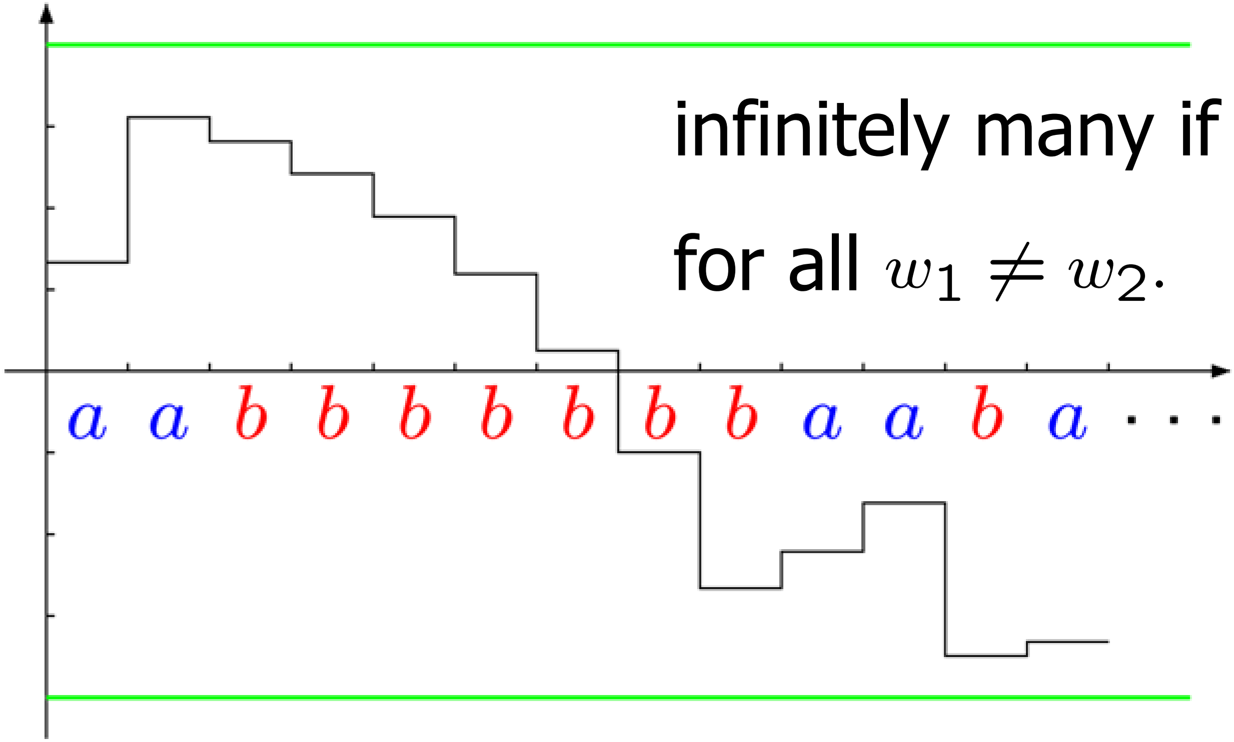
How many different values can $diff(w)$ take ?



Disc_λ 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

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

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

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

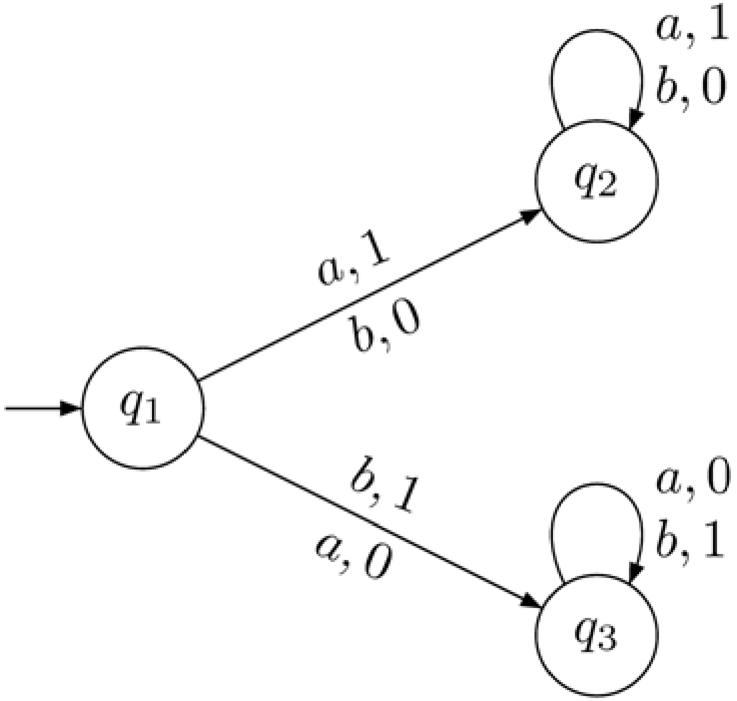
for all $w_1 \neq w_2$.

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.

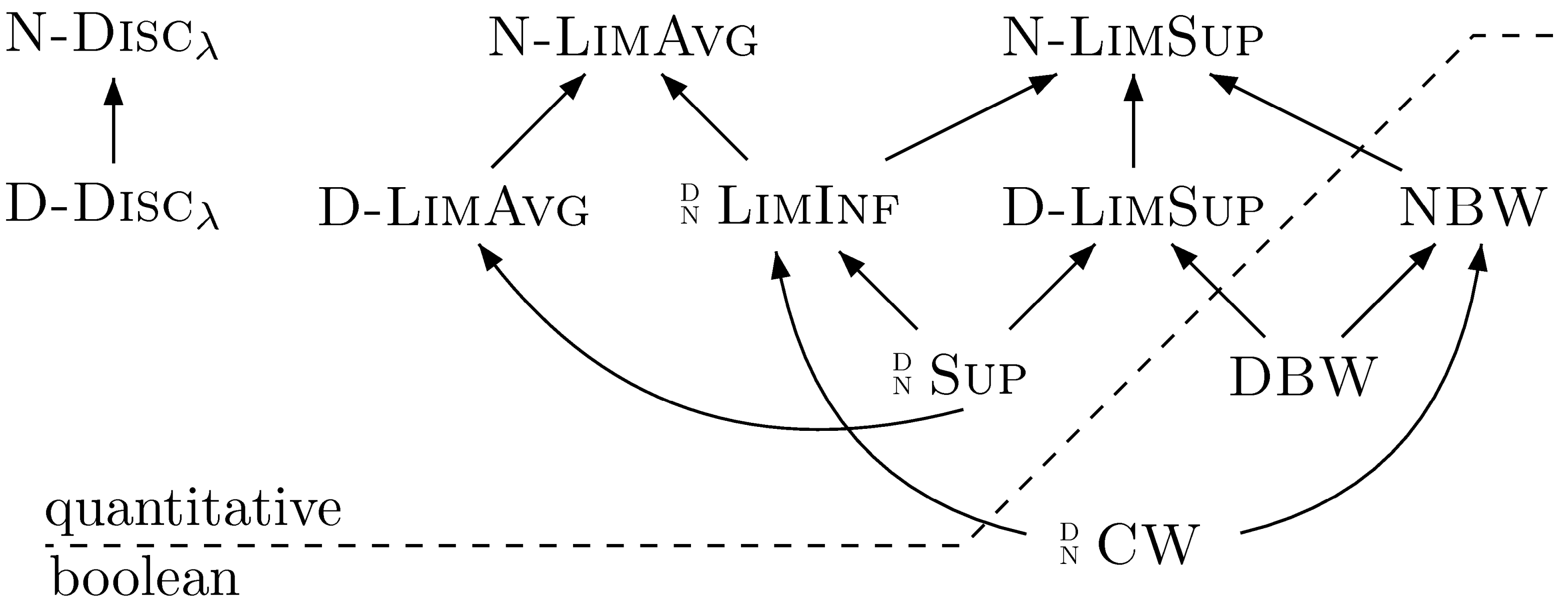
Last result

Disc_λ cannot be determinized.



λ=3/4

Reducibility relations



Conclusion

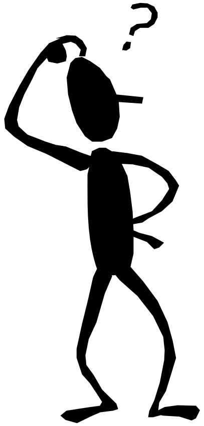
- Quantitative generalization of languages to model programs/systems more accurately.

- 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 $Disc_\lambda$ cannot be determinized.

The end

Thank you !



Questions ?