# Minimization of Weighted Automata

Andreas Maletti

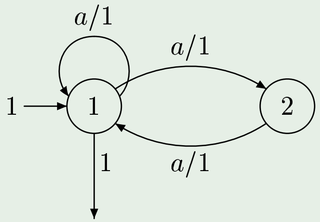
International Computer Science Institute
Berkeley, CA, USA

Dresden — May 14, 2008

# Table of Contents

# Syntax

**Example**




**Definition (Weighted automaton)**

$(Q, \Sigma, \mu, I, F)$

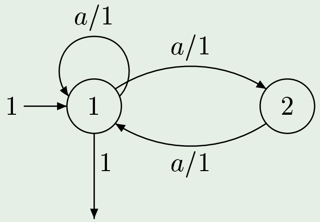
- $Q$ finite set of states
- $\Sigma$ alphabet
- $\mu \colon Q \times \Sigma \times Q \to A$
- $I \colon Q \to A$ initial weights
- $F \colon Q \to A$ final weights

**References**

- Berstel, Reutenauer: Rational series and their languages. Springer, 1988
- Kuich, Salomaa: Semirings, automata, languages. Springer, 1986

# Syntax

## Example




## Definition (Weighted automaton)

$(Q, \Sigma, \mu, I, F)$

- $Q$ finite set of states
- $\Sigma$ alphabet
- $\mu \colon Q \times \Sigma \times Q \to A$
- $I \colon Q \to A$ initial weights
- $F \colon Q \to A$ final weights

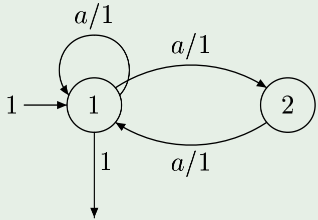
## References


- Berstel, Reutenauer: Rational series and their languages. Springer, 1988
- Kuich, Salomaa: Semirings, automata, languages. Springer, 1986

# Semantics

Weight structure: Semiring $A = (A, +, \cdot, 0, 1)$

**Example**




**Definition (Semantics)**

$h_\mu \colon \Sigma^* \to A^Q$

$$h_\mu(\varepsilon)_q = I(q)$$

$$h_\mu(wa)_q = \sum_{p \in Q} h_\mu(w)_p \cdot \mu(p, a, q)$$

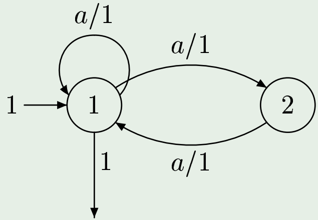
**Example (using natural numbers)**

$h_\mu(a) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ $h_\mu(aa) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$ $h_\mu(aaa) = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$

# Semantics

Weight structure: Semiring $A = (A, +, \cdot, 0, 1)$

## Example




## Definition (Semantics)

$h_\mu \colon \Sigma^* \to A^Q$

$$h_\mu(\varepsilon)_q = I(q)$$

$$h_\mu(wa)_q = \sum_{p \in Q} h_\mu(w)_p \cdot \mu(p, a, q)$$

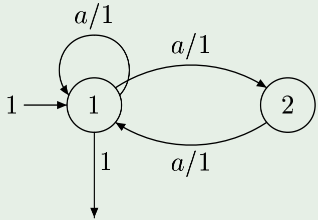
## Example (using natural numbers)

$$h_\mu(a) = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad h_\mu(aa) = \begin{pmatrix} 2 \\ 1 \end{pmatrix} \qquad h_\mu(aaa) = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

# Semantics

Weight structure: Semiring $A = (A, +, \cdot, 0, 1)$

## Example




## Definition (Semantics)

$h_\mu \colon \Sigma^* \to A^Q$

$$h_\mu(\varepsilon)_q = I(q)$$

$$h_\mu(wa)_q = \sum_{p \in Q} h_\mu(w)_p \cdot \mu(p, a, q)$$

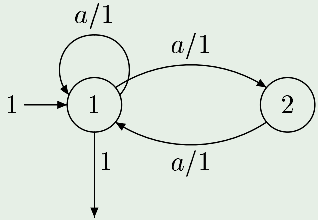
## Example (using natural numbers)

$$h_\mu(a) = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad h_\mu(aa) = \begin{pmatrix} 2 \\ 1 \end{pmatrix} \quad h_\mu(aaa) = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

# Semantics

Weight structure: Semiring $A = (A, +, \cdot, 0, 1)$

## Example




## Definition (Semantics)

$h_\mu \colon \Sigma^* \to A^Q$

$$h_\mu(\varepsilon)_q = I(q)$$

$$h_\mu(wa)_q = \sum_{p \in Q} h_\mu(w)_p \cdot \mu(p, a, q)$$

## Example (using natural numbers)

$$h_\mu(a) = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad h_\mu(aa) = \begin{pmatrix} 2 \\ 1 \end{pmatrix} \quad h_\mu(aaa) = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

# Table of Contents

# Overview

## Results

| Method | Nondet. | Det. | Complexity | Reference |
|---|---|---|---|---|
| Pushing & HOPCROFT | – | x | $O(m \log n)$ | Mohri |
| Forward Bisimulation | x | x | $O(m \log n)$ | Buchholz |
| Backward Bisimulation | x | – | $O(m \log n)$ | Buchholz |
| Backward Simulation | x | – | $O(mn)$ | Ranzato, ... |
| Full minimization | x | x | $\mathrm{P}$ | Berstel, ... |

## Notation

- $m$: number of transitions
- $n$: number of states

# Pushing & HOPCROFT [Mohri]

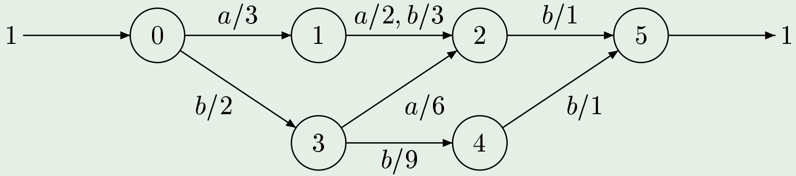
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

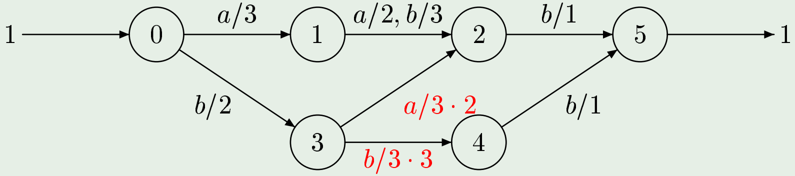
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

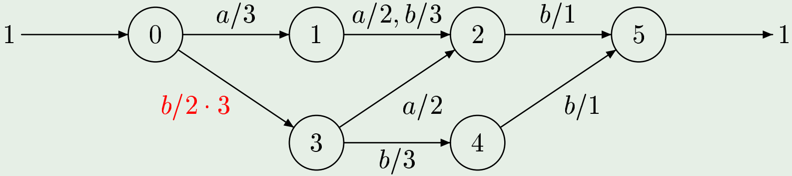
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

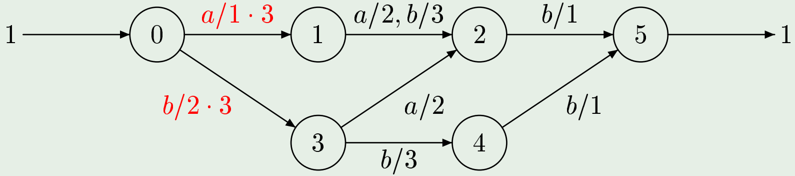
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

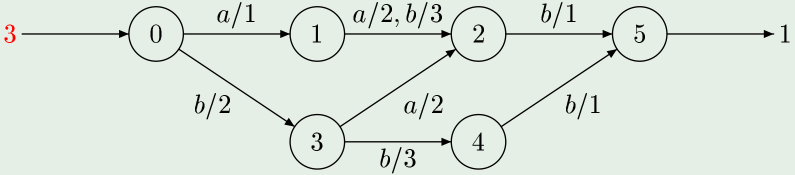
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

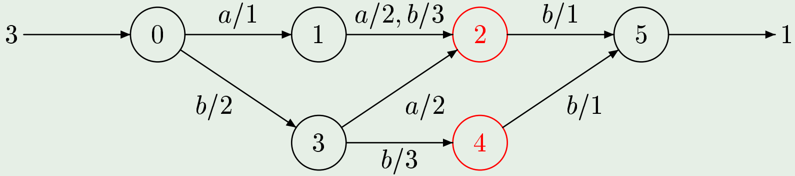
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

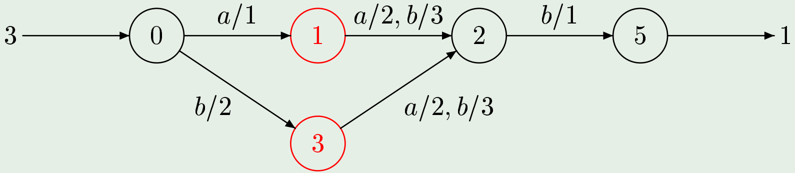
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

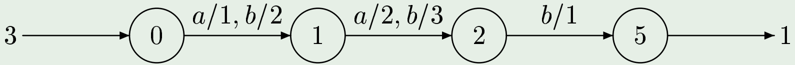
## 1. Push

Move weights toward the front

## 2. Minimize

Minimize as unweighted automaton; treat weight as part of label

## Example (ala Eisner)



# Pushing & HOPCROFT [Mohri]

## Prerequisites (ala Eisner)

1. Automaton deterministic
2. Semiring multiplicatively cancellative
3. Semiring allows greedy factorization

## Definition (Greedy factorization)

There exists a mapping $f \colon A^2 \to A$ such that for all $a, b, c \in A$ with $c \neq 0$:

$$\text{If } a|c \text{ and } b|c \text{ , then } \frac{c}{a \cdot f(a,b)} = \frac{c}{b \cdot f(b,a)}$$

# Pushing & HOPCROFT [Mohri]

## Prerequisites (ala Eisner)

1. Automaton deterministic
2. Semiring multiplicatively cancellative
3. Semiring allows greedy factorization

## Definition (Greedy factorization)

There exists a mapping $f \colon A^2 \to A$ such that for all $a, b, c \in A$ with $c \neq 0$:

$$\text{If } a|c \text{ and } b|c \text{ , then } \frac{c}{a \cdot f(a, b)} = \frac{c}{b \cdot f(b, a)}$$

# Forward Bisimulation [Buchholz]

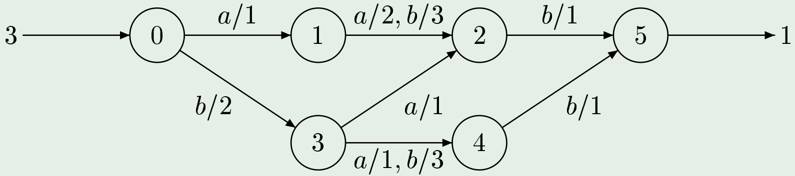
## Definition (Forward bisimulation)

Equivalence relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\sum_{r \in [q]} \mu(p, a, r) = \sum_{r \in [q]} \mu(p', a, r)$$

for every $p \equiv p'$, state $q$, and symbol $a$.

## Example



# Forward Bisimulation [Buchholz]

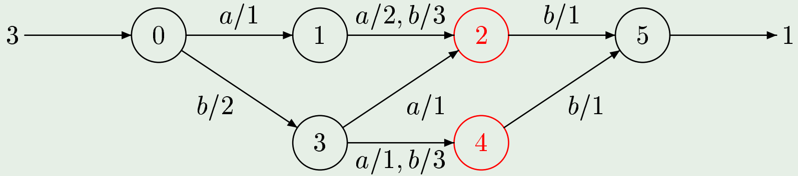
## Definition (Forward bisimulation)

Equivalence relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\sum_{r \in [q]} \mu(p, a, r) = \sum_{r \in [q]} \mu(p', a, r)$$

for every $p \equiv p'$, state $q$, and symbol $a$.

## Example



# Forward Bisimulation [Buchholz]

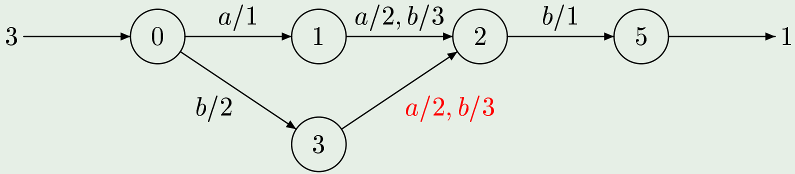
## Definition (Forward bisimulation)

Equivalence relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\sum_{r \in [q]} \mu(p, a, r) = \sum_{r \in [q]} \mu(p', a, r)$$

for every $p \equiv p'$, state $q$, and symbol $a$.

## Example



# Forward Bisimulation [Buchholz]

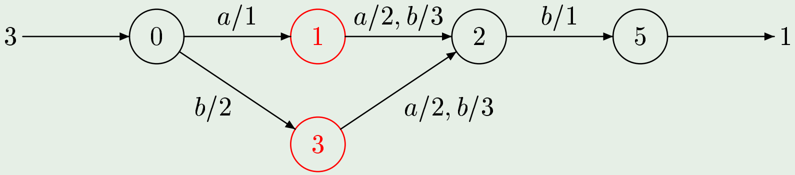
## Definition (Forward bisimulation)

Equivalence relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\sum_{r \in [q]} \mu(p, a, r) = \sum_{r \in [q]} \mu(p', a, r)$$

for every $p \equiv p'$, state $q$, and symbol $a$.

## Example



# Forward Bisimulation [Buchholz]

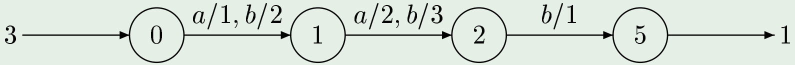
## Definition (Forward bisimulation)

Equivalence relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\sum_{r \in [q]} \mu(p, a, r) = \sum_{r \in [q]} \mu(p', a, r)$$

for every $p \equiv p'$, state $q$, and symbol $a$.

## Example



# Unweighted Forward Simulation [Abdulla et. al.]

## Definition (Forward bisimulation unweighted)

Reflexive, symmetric, and transitive relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\exists q' \equiv q \colon \mu(p, a, q) \leq \mu(p', a, q')$$

for every $p \equiv p'$, state $q$, and symbol $a$.

# Unweighted Forward Simulation [Abdulla et. al.]

## Definition (Forward bisimulation unweighted)

Reflexive, symmetric, and transitive relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\exists q' \equiv q \colon \mu(p, a, q) \leq \mu(p', a, q')$$

for every $p \equiv p'$, state $q$, and symbol $a$.

# Unweighted Forward Simulation [Abdulla et. al.]

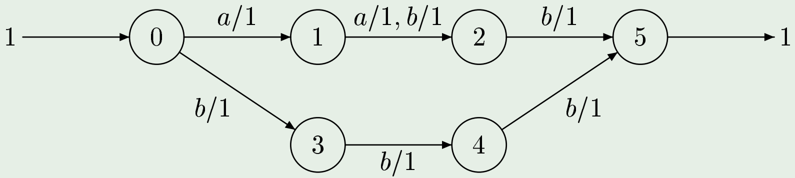
## Definition (Forward simulation)

Pre-order $\leq$ on states such that $F(p) \leq F(p')$ and

$$\exists q' \geq q \colon \mu(p, a, q) \leq \mu(p', a, q')$$

for every $p \leq p'$, state $q$, and symbol $a$.

## Example




$0 \quad 3 \leq 1 \quad 4 \leq 2 \leq 4 \quad 5$

# Unweighted Forward Simulation [Abdulla et. al.]

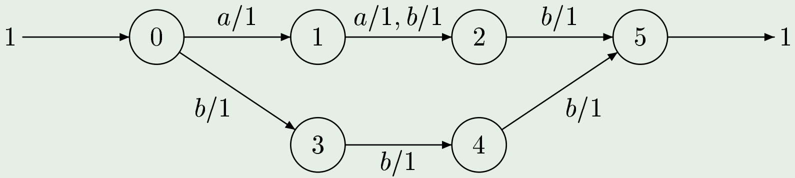
## Definition (Forward simulation)

Pre-order $\leq$ on states such that $F(p) \leq F(p')$ and

$$\exists q' \geq q \colon \mu(p, a, q) \leq \mu(p', a, q')$$

for every $p \leq p'$, state $q$, and symbol $a$.

## Example




$0 \quad 3 \leq 1 \quad 4 \leq 2 \leq 4 \quad 5$

# Unweighted Forward Simulation [Abdulla et. al.]

## Definition (Forward simulation)

Pre-order $\leq$ on states such that $F(p) \leq F(p')$ and

$$\exists q' \geq q : \mu(p, a, q) \leq \mu(p', a, q')$$

for every $p \leq p'$, state $q$, and symbol $a$.

## Note

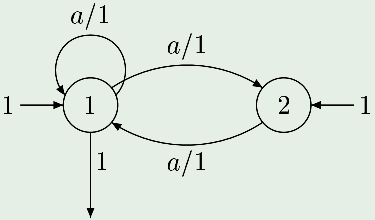
Does in general not preserve the language!

# Backward Simulation [Abdulla et. al.]

**Definition (Backward (bi)simulation)**

A forward (bi)simulation on the reversed automaton.

**Example**




$1 \leq 2 \leq 1$

**Theorem**

Reducing automaton by $\leq \cap \geq$ with $\leq$ a backward simulation preserves the language.

**Note**

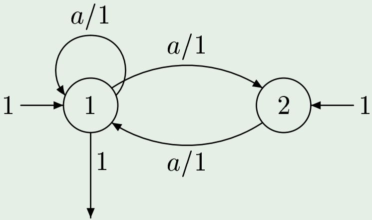
Slightly more general than backward bisimulation.

# Backward Simulation [Abdulla et. al.]

**Definition (Backward (bi)simulation)**

A forward (bi)simulation on the reversed automaton.

**Example**




$1 \leq 2 \leq 1$

**Theorem**

Reducing automaton by $\leq \cap \geq$ with $\leq$ a backward simulation preserves the language.

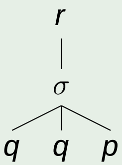
**Note**

Slightly more general than backward bisimulation.

# Table of Contents

# Syntax

## Example (Transition)




## Definition (Weighted tree automaton)

$(Q, \Sigma, (\mu_k)_{k \in \mathbb{N}}, F)$

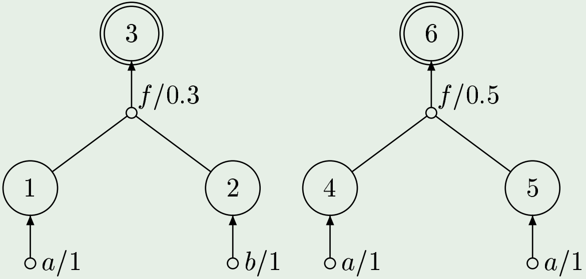
- $Q$ finite set of states
- $\Sigma$ ranked alphabet
- $\mu_k : Q^k \times \Sigma_k \times Q \to A$
- $F : Q \to A$ final weights

## References

- Berstel, Reutenauer: Recognizable formal power series on trees. TCS 18, 1982
- Borchardt: The theory of recognizable tree series. Dissertation, 2004

# Syntax — Illustration

## Example



# Semantics

## Definition

Let $t \in T_\Sigma(Q)$ and $W = \text{pos}(t)$.

- Run on $t$: map $r\colon W \to Q$ with $r(w) = t(w)$ if $t(w) \in Q$
- Weight of $r$

$$\text{wt}(r) = \prod_{\substack{w \in W \\ t(w) \in \Sigma}} \mu_k(r(w1), \ldots, r(wk), t(w), r(w))$$

- Recognized tree series

$$(\|M\|, t) = \sum_{r \text{ run on } t} F(r(\varepsilon)) \cdot \text{wt}(r)$$

# Semantics

## Definition

Let $t \in T_\Sigma(Q)$ and $W = \text{pos}(t)$.

- **Run** on $t$: map $r\colon W \to Q$ with $r(w) = t(w)$ if $t(w) \in Q$
- **Weight** of $r$

$$\text{wt}(r) = \prod_{\substack{w \in W \\ t(w) \in \Sigma}} \mu_k(r(w1), \ldots, r(wk), t(w), r(w))$$

- Recognized tree series

$$(\|M\|, t) = \sum_{r \text{ run on } t} F(r(\varepsilon)) \cdot \text{wt}(r)$$

# Semantics

## Definition

Let $t \in T_\Sigma(Q)$ and $W = \text{pos}(t)$.

- **Run** on $t$: map $r \colon W \to Q$ with $r(w) = t(w)$ if $t(w) \in Q$
- **Weight** of $r$

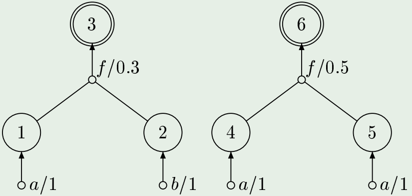
$$\text{wt}(r) = \prod_{\substack{w \in W \\ t(w) \in \Sigma}} \mu_k(r(w1), \dots, r(wk), t(w), r(w))$$

- **Recognized tree series**

$$(\|M\|, t) = \sum_{r \text{ run on } t} F(r(\varepsilon)) \cdot \text{wt}(r)$$

# Semantics — Illustration

## Example (Automaton)



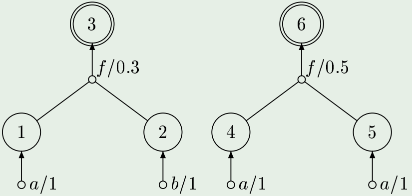

## Example (Runs)

Input tree: $f(a, b)$

Runs: $6(1, 2)$ with weight 0

# Semantics — Illustration

## Example (Automaton)

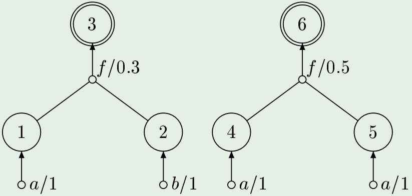



## Example (Runs)

Input tree: $f(a, b)$ Runs: $f(a, b)$ with weight

# Semantics — Illustration

## Example (Automaton)



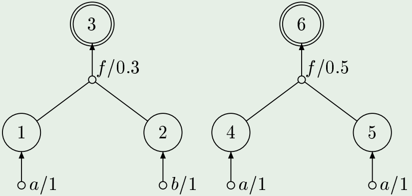

## Example (Runs)

Input tree: $f(a, b)$

Runs: $f(1, b)$ with weight 1

# Semantics — Illustration

## Example (Automaton)

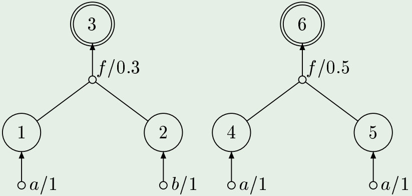



## Example (Runs)

Input tree: $f(a, b)$ Runs: $f(1, 2)$ with weight 1

# Semantics — Illustration

## Example (Automaton)




## Example (Runs)

Input tree: $f(a, b)$

Runs: 3 with weight 0.3

$3(1, 2)$

# Table of Contents

# Overview

## Results

| Method | Nondet. | Det. | Complexity | Reference |
|---|---|---|---|---|
| Det. Minimization | – | x | $O(rmn^4)$ | |
| Forw. Bisimulation | x | x | $O(rm \log n)$ | Högberg, ... |
| Backw. Bisimulation | x | – | $O(r^2 m \log n)$ | Högberg, ... |
| Backw. Simulation | x | – | $O(r^2 mn)$ | Abdulla, ... |
| Full minimization | x | x | $\mathrm{P}$ | Bozapalidis |

## Notation

- $m$: number of transitions
- $n$: number of states
- $r$: maximal rank of the input symbols

# Forward Bisimulation [Högberg, et. al.]

## Definition (Forward bisimulation)

Equivalence relation $\equiv$ on states such that $F(p) = F(p')$ and

$$\sum_{r \in [q]} \mu(\ldots, p, \ldots, a, r) = \sum_{r \in [q]} \mu(\ldots, p', \ldots, a, r)$$

for every $p \equiv p'$, symbol $a$, and states $q$ and $\ldots$.

# Backward Bisimulation [Högberg, et. al.]

**Definition (Backward bisimulation)**

Equivalence relation $\equiv$ on states such that

$$\sum_{q_1 \cdots q_k \in B_1 \cdots B_k} \mu(q_1, \ldots, q_k, a, p) = \sum_{q_1 \cdots q_k \in B_1 \cdots B_k} \mu(q_1, \ldots, q_k, a, p')$$

for every $p \equiv p'$, symbol $a$, and blocks $B_1, \ldots, B_k$.

# Det. Minimization — Overview

## Applicability

- Deterministic wta
- Commutative semifield (i.e. multiplicative inverses)

## Roadmap

- Myhill-Nerode congruence relation [Borchardt]
- Determine signs of life
- Refinement

# MYHILL-NERODE congruence

## Definition

$p \equiv q$: there exists nonzero $a$ such that for every context $C$

$$(\|M\|, C[p]) = a \cdot (\|M\|, C[q])$$

## Notes

- Semifields are zero-divisor free
- Element $a$ is unique if $p$ is not dead

# MYHILL-NERODE congruence

## Definition

$p \equiv q$: there exists nonzero $a$ such that for every context $C$

$$(\|M\|, C[p]) = a \cdot (\|M\|, C[q])$$

## Notes

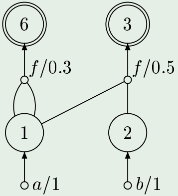
- Semifields are zero-divisor free
- Element $a$ is unique if $p$ is not dead

# Signs of Life

## Definition

Sign of life of $q \in Q$: context $C$ such that $(\|M\|, C[q]) \neq 0$

## Example




| State | Sign of life | State | Sign of life |
|---|---|---|---|
| 1 | $f(\square, b)$ | 2 | $f(1, \square)$ |
| 3 | $\square$ | 6 | $\square$ |

# Stages

## Definition

Stage $(\Pi, \text{sol}, f, r)$:

(i) $\equiv$ refinement of $\equiv_\Pi$

(ii) $\text{sol}(F) = \{\Box\}$

(iii) for live $q$ with $p = r([q])$

$$(\|M\|, \text{sol}(p)[q]) = f(q) \cdot (\|M\|, \text{sol}(p)[p])$$

(iv) $\equiv_\Pi$ congruence

(v) for symbol $\sigma$ and context $C$ with live $\delta_\sigma(C[q])$

$$f(q)^{-1} \cdot c_\sigma(C[q]) \cdot f(\delta_\sigma(C[q])) = c_\sigma(C[p]) \cdot f(\delta_\sigma(C[p]))$$

where $\delta_\sigma : Q^k \to Q$ and $c_\sigma : Q^k \to A$

# Stages

## Definition

Stable stage $(\Pi, \text{sol}, f, r)$:

(i) $\equiv$ refinement of $\equiv_\Pi$

(ii) $\text{sol}(F) = \{\Box\}$

(iii) for live $q$ with $p = r([q])$

$$(\|M\|, \text{sol}(p)[q]) = f(q) \cdot (\|M\|, \text{sol}(p)[p])$$

(iv) $\equiv_\Pi$ congruence

(v) for symbol $\sigma$ and context $C$ with live $\delta_\sigma(C[q])$

$$f(q)^{-1} \cdot c_\sigma(C[q]) \cdot f(\delta_\sigma(C[q])) = c_\sigma(C[p]) \cdot f(\delta_\sigma(C[p]))$$

where $\delta_\sigma : Q^k \to Q$ and $c_\sigma : Q^k \to A$

# Refining a Stage

## Definition

Refinement of $(\Pi, \text{sol}, f, r)$: Partition $\Pi'$ with $p \equiv_{\Pi'} q$ if

(i) $p \equiv_\Pi q$

(ii) $\delta_\sigma(C[p]) \equiv_\Pi \delta_\sigma(C[q])$

(iii) if $\delta_\sigma(C[p])$ is live, then

$$f(p)^{-1} \cdot c_\sigma(C[p]) \cdot f(\delta_\sigma(C[p])) = f(q)^{-1} \cdot c_\sigma(C[q]) \cdot f(\delta_\sigma(C[q]))$$

for states $p$ and $q$, symbol $\sigma$, and context $C$

# Complete Algorithm

## Algorithm

$(\Pi', \mathrm{sol}, D) \leftarrow \textsc{ComputeSol}(M)$
**repeat**
  $(\Pi, \mathrm{sol}, f, r) \leftarrow \textsc{Complete}(M, \Pi', \mathrm{sol}, D)$
  $\Pi' \leftarrow \textsc{Refine}(M, \Pi, \mathrm{sol}, f, r, D)$
**until** $\Pi' = \Pi$
**return** minimized wta

## Notes

- Algorithm runs in $O(rmn^4)$
- Returns equivalent minimal deterministic wta

# Complete Algorithm

## Algorithm

$(\Pi', \text{sol}, D) \leftarrow \textsc{ComputeSol}(M)$
**repeat**
  $(\Pi, \text{sol}, f, r) \leftarrow \textsc{Complete}(M, \Pi', \text{sol}, D)$
  $\Pi' \leftarrow \textsc{Refine}(M, \Pi, \text{sol}, f, r, D)$
**until** $\Pi' = \Pi$
**return** minimized wta

## Notes

- Algorithm runs in $O(rmn^4)$
- Returns equivalent minimal deterministic wta

# Table of Contents

# Experiments

## State Count

| Original | Minimal | Reduction to |
| ---: | ---: | ---: |
| 98 | 68 | 69% |
| 394 | 308 | 78% |
| 497 | 381 | 77% |
| 727 | 515 | 71% |
| 2701 | 1993 | 74% |
| 3686 | 1766 | 48% |

## State & Transition Count

| Error | Original | Minimal | Reduction to |
| --- | --- | --- | --- |
| $10^{-4}$ | $(727, 6485)$ | $(629, 6131)$ | $(87\%, 95\%)$ |
| $10^{-2}$ | $(727, 6485)$ | $(525, 3425)$ | $(72\%, 53\%)$ |

# Experiments

## State Count

| Original | Minimal | Reduction to |
|---|---|---|
| 98 | 68 | 69% |
| 394 | 308 | 78% |
| 497 | 381 | 77% |
| 727 | 515 | 71% |
| 2701 | 1993 | 74% |
| 3686 | 1766 | 48% |

## State & Transition Count

| Error | Original | Minimal | Reduction to |
|---|---|---|---|
| $10^{-4}$ | $(727, 6485)$ | $(629, 6131)$ | $(87\%, 95\%)$ |
| $10^{-2}$ | $(727, 6485)$ | $(525, 3425)$ | $(72\%, 53\%)$ |

# The End

Thank You!