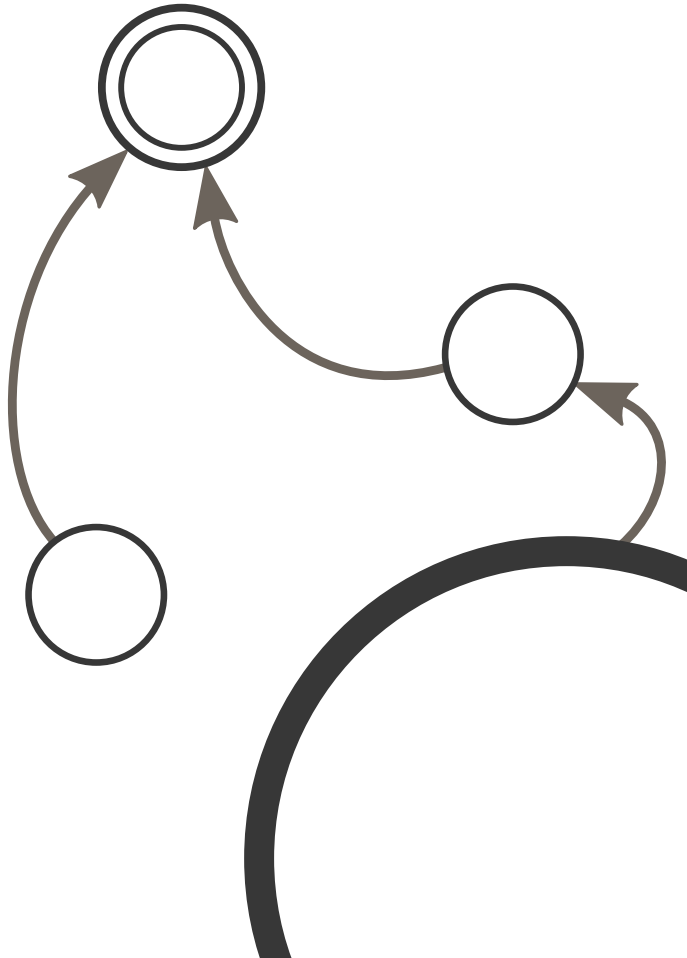
COMPUTATIONAL LIMITATIONS OF AFFINE AUTOMATA

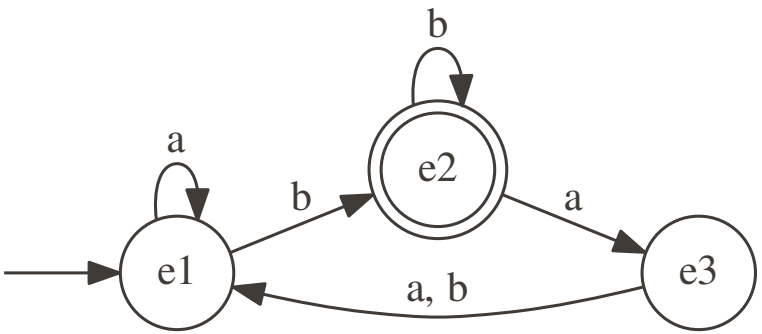
UCNC2019
Tokyo, June 7, 2019

Mika Hirvensalo, **Etienne Moutot**
and Abuzer Yakaryılmaz



DEFINITIONS, MOTIVATION

DETERMINISTIC FINITE AUTOMATON



A Deterministic Finite Automaton

$$(\mathbf{M}_x)_{i,j} = 1 \text{ iff } j \xrightarrow{x} i$$

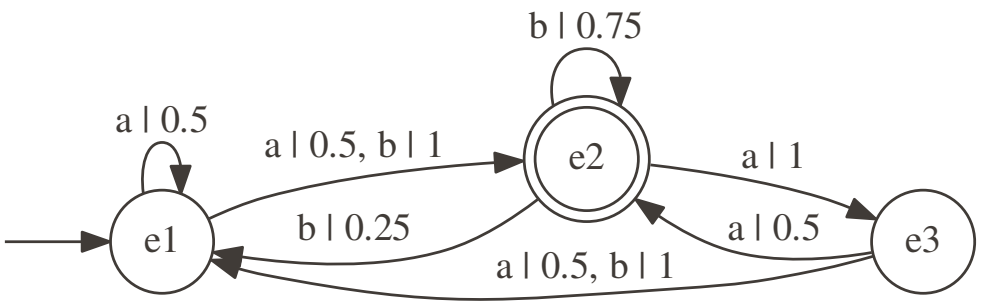
$$\mathbf{M}_a = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{M}_b = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

PROBABILISTIC FINITE AUTOMATON

$$(\mathbf{M}_x)_{i,j} = P(j \xrightarrow{x} i)$$

$\mathbf{M}_x$ are stochastic matrices



A Probabilistic Finite Automaton

$$\mathbf{M}_a = \begin{pmatrix} 0.5 & 0 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{M}_b = \begin{pmatrix} 0 & 0.25 & 1 \\ 1 & 0.75 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

PROBABILISTIC FINITE AUTOMATON

$w = x_1 \dots x_n$

$\mathbf{M}_w = \mathbf{M}_{x_n} \dots \mathbf{M}_{x_1}$

P projection associated to accepting states

$\mathbf{v}_0$ initial vector

$$f_P(w) = \mathbf{P}\,\mathbf{M}_w\,\mathbf{v}_0$$

STOCHASTIC LANGUAGES

Definition (Stochastic language - SL)

$$L = \{w \in \Sigma^* \mid f_P(w) > \lambda\}$$

$\lambda \in (0, 1)$ is called the **cutpoint**

STOCHASTIC LANGUAGES

Definition (Isolated cutpoint - ISL)

$\lambda \in (0,1)$ is an **isolated cutpoint** iff there exist $\delta > 0$ such that

- $\forall w \in L, f_P(w) \geq \lambda + \delta$
- $\forall w \notin L, f_P(w) \leq \lambda - \delta$

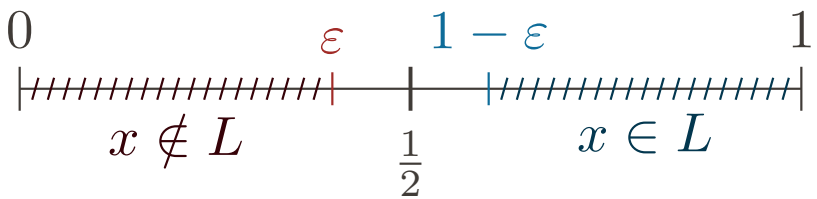
STOCHASTIC LANGUAGES

Definition (Bounded error - BSL)

L is recognized with **bounded error** iff there exists $\varepsilon \in (0, \frac{1}{2})$ such that

$\rightarrow \forall w \in L, f_P(w) \geq 1 - \varepsilon$

$\rightarrow \forall w \notin L, f_P(w) \leq \varepsilon$



STOCHASTIC LANGUAGES

Theorem

$\mathrm{ISL} = \mathrm{BSL} = \mathrm{REG}$

GENERAL AUTOMATA ?

Theorem (Jeandel, 2007)

The bounded-error language of any topological automata with compact set of states and continuous evaluation function is REG.

AFFINE FINITE AUTOMATON

$$A = (E, \Sigma, \{\mathbf{M}_x \mid x \in \Sigma\}, \mathbf{v}_0, E_a)$$

E	finite set of states of P
Σ	finite alphabet
$\{\mathbf{M}_x\}$	set of transition matrices, all columns sums up to 1
$\mathbf{v}_0$	initial vector, coordinates sums up to 1
E_a	set of accepting states

AFFINE FINITE AUTOMATON

$w = x_1 \dots x_n$

$\mathbf{M}_w = \mathbf{M}_{x_n} \dots \mathbf{M}_{x_1}$

P projection associated to accepting states

$\mathbf{v}_0$ initial vector

$$f_A(w) = \frac{|\mathbf{P}\mathbf{M}_w\mathbf{v}_0|}{|\mathbf{M}_w\mathbf{v}_0|} = \frac{\sum_{e_i \in E_a} |(\mathbf{M}_w\mathbf{v}_0)_i|}{\sum_{e_i \in E} |(\mathbf{M}_w\mathbf{v}_0)_i|}$$

AFFINE LANGUAGE

Definition (Affine language - AfL)

$$L = \{w \in \Sigma^* \mid f_P(w) > \lambda\}$$

$\lambda \in (0, 1)$ is called the **cutpoint**

BOUNDED ERROR

Definition (Isolated cutpoint - IAfL)

$\lambda \in (0,1)$ is an **isolated cutpoint** iff there exist $\delta > 0$ such that

$\rightarrow \forall w \in L, f_P(w) \geq \lambda + \delta$

$\rightarrow \forall w \notin L, f_P(w) \leq \lambda - \delta$

Definition (Bounded error - BAfL)

L is recognized with **bounded error** iff there exists $\varepsilon \in (0, \frac{1}{2})$ such that

$\rightarrow \forall w \in L, f_P(w) \geq 1 - \varepsilon$

$\rightarrow \forall w \notin L, f_P(w) \leq \varepsilon$

AFFINE LANGUAGES

Theorem (Díaz-Caron, Yakaryilmaz, 2016)

$\mathrm{IAfL} = \mathrm{BAfL}$

AFFINE LANGUAGES

Theorem (Díaz-Caron, Yakaryilmaz, 2016)

$\mathsf{IAfL} = \mathsf{BAfL} \neq \mathsf{REG}$

OPERATION ON LANGUAGES

Theorem (Hirvensalo, M. & Yakaryılmaz, 2017)

BAfL *is stable under union, intersection and complement.*

AFFINE AUTOMATA SIMULATION

LOGARITHMIC SIMULATION (STOCHASTIC CASE)

Theorem (Macarie 98)

$SL_{\mathbb{Q}} \subseteq L$ (Stochastic automata can be simulated in logspace)

LOGARITHMIC SIMULATION

Theorem (Hirvensalo, M. & Yakaryılmaz)

$\mathrm{AfL}_{\mathbb{Q}} \subseteq \mathrm{L}$ (Affine automata can be simulated in logspace)

MACARIE KEY IDEAS: MODULAR ARITHMETICS

Residue representation of $\mathbf{x}$ w.r.t $\mathbf{n}$:

$$\mathbf{x} \mod \mathbf{n} = (\mathbf{x} \mod n_1, \ldots, \mathbf{x} \mod n_r)$$

Chinese Remainder Theorem:

1. $\mathbf{n} = (n_1, \ldots, n_r)$ consists of pairwise coprime integers
2. $\forall i, x_i \leq N - 1$ with $N = n_1 \cdots n_r$

Then $\mathbf{x}$ **can be recovered from its residue representation**

MACARIE KEY IDEAS: MODULAR ARITHMETICS

Complex operations on $\mathbf{x}$

$\rightarrow$ Do operations in $O(\log x)$ space

MACARIE KEY IDEAS: MODULAR ARITHMETICS

Complex operations on $\mathbf{x} \mod p$

$\rightarrow$ Do operations in $O(\log p)$ space

MACARIE KEY IDEAS: MODULAR ARITHMETICS

Complex operations on $\mathbf{x} \mod p$

$\rightarrow$ Do operations in $O(\log p)$ space

Prime number theorem:

$$P_r = 3 \cdot 5 \cdot 7 \cdot \ \cdots \ \cdot p_r = \frac{1}{2} e^{(1+o(1))r \ln r}$$

$\rightarrow$ Do operations in $O(\log r)$ space on integers of size $\frac{1}{2} e^{(1+o(1))r \ln r}$

MACARIE KEY IDEAS: MODULAR ARITHMETICS

$\mathbf{p}_r = (3, 5, 7, \ldots, p_r)$
$P_r = 3 \cdot 5 \cdot 7 \cdot \ \cdots \ \cdot p_r$

Lemma (Macarie)

Given the residue representations of integers $x, y \in [0, P_r - 1]$,

Decisions $x > y$, $x = y$ or $x < y$ take $O(\log p_r)$ space.

USING MACARIE'S LEMMA

$$L = \left\{ w \in \Sigma^* \mid f_P(w) > \frac{1}{2} \right\}$$

$$= \left\{ w \in \Sigma^* \mid 2\mathbf{y}^T D M_{w_n} \cdots D M_{w_1} \mathbf{x} > D^n \right\}$$

Both side need only $O(n)$ primes $\rightarrow$ Comparison in $O(\log n)$ space

USING MACARIE'S LEMMA

$$L = \left\{ w \in \Sigma^* \mid f_P(w) > \frac{1}{2} \right\}$$

$$= \left\{ w \in \Sigma^* \mid 2\mathbf{y}^T D M_{w_n} \cdots D M_{w_1} \mathbf{x} > D^n \right\}$$

Both side need only $O(n)$ primes $\rightarrow$ Comparison in $O(\log n)$ space

Theorem (Macarie 98)

$\mathsf{SL}_{\mathbb{Q}} \subseteq \mathsf{L}$

AFFINE CASE: ABSOLUTE VALUES

Problem in the affine case: **the absolute values**

$$2 \equiv -3 \mod 5$$

But

$$|2| \not\equiv |-3| \mod 5$$

REMOVING ABSOLUTE VALUES: FIRST TRY

$$L = \left\{ w \in \Sigma^* \mid \frac{|\mathbf{P}\mathbf{M}_w\mathbf{v}_0|}{|\mathbf{M}_w\mathbf{v}_0|} > \frac{1}{2} \right\}$$
$$= \{ w \in \Sigma^* \mid 2|\mathbf{P}\mathbf{M}_w\mathbf{v}_0| > |\mathbf{M}_w\mathbf{v}_0| \}$$

Ideal case:

Positive matrix values $\Rightarrow$ no absolute values !

REMOVING ABSOLUTE VALUES: FIRST TRY

Ideal case:
Positive matrix values $\Rightarrow$ no absolute values !

→ Very big m

→ $\mathbb{E} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{pmatrix}$

$$C_i = M_i + m\mathbb{E}$$

REMOVING ABSOLUTE VALUES: FIRST TRY

Ideal case:
Positive matrix values $\Rightarrow$ no absolute values !

→ Very big m

→ $\mathbb{E} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{pmatrix}$

$$C_i = M_i + m\mathbb{E}$$

$$C_{ab} = (M_a + m\mathbb{E})(M_b + m\mathbb{E}) = M_{ab} + m^2\mathbb{E}^2 + mM_a\mathbb{E} + mM_b\mathbb{E}$$

REMOVING ABSOLUTE VALUES: TURAKAINEN'S TRICK

$$B_i = \begin{pmatrix} 0 & \mathbf{0}^T & 0 \\ \mathbf{c}_i & M_i & \mathbf{0} \\ e_i & \mathbf{d}_i^T & 0 \end{pmatrix}$$

$\mathbf{c}_i$, $\mathbf{d}_i$, e_i s.t. column and row of B_i sum to zero

$$B_i\mathbb{E} = \mathbb{E}B_i = \mathbf{0}$$

$$C_i = B_i + m\mathbb{E}$$

$$C_w = B_w + m^{|w|}(k+2)^{|w|-1}\mathbb{E}$$

REMOVING ABSOLUTE VALUES: TURAKAINEN'S TRICK

$$\frac{|FM_w\mathbf{x}|}{|M_w\mathbf{x}|} > \frac{1}{2}$$

$$\Leftrightarrow$$

$$2\left|F'C_w\mathbf{v}'_0 - m^{|w|}(k+2)^{|w|-1}F'\mathbb{E}\mathbf{x}'\right| > \left|C_w\mathbf{x}' - m^{|w|}(k+2)^{|w|-1}\mathbb{E}\mathbf{x}'\right|$$

→ Residue of $\mathbf{a} = F'C_w\mathbf{v}'_0$: $O(\log n)$ (Macarie)

→ Residue of $\mathbf{b})m^{|w|}(k+2)^{|w|-1}F'\mathbb{E}\mathbf{x}'$: $O(\log n)$

→ $|\mathbf{a}-\mathbf{b}| = |\mathbf{a}_1 - \mathbf{b}_1| + \cdots + |\mathbf{a_k} - \mathbf{b}_k|$

REMOVING ABSOLUTE VALUES: TURAKAINEN'S TRICK

$$\frac{|FM_w\mathbf{x}|}{|M_w\mathbf{x}|} > \frac{1}{2}$$

$$\Leftrightarrow$$

$$2\left|F'C_w\mathbf{v}'_0 - m^{|w|}(k+2)^{|w|-1}F'\mathbb{E}\mathbf{x}'\right| > \left|C_w\mathbf{x}' - m^{|w|}(k+2)^{|w|-1}\mathbb{E}\mathbf{x}'\right|$$

- → Residue of $\mathbf{a} = F'C_w\mathbf{v}'_0$: $O(\log n)$ (Macarie)
- → Residue of $\mathbf{b})m^{|w|}(k+2)^{|w|-1}F'\mathbb{E}\mathbf{x}'$: $O(\log n)$
- → $|\mathbf{a}-\mathbf{b}| = |\mathbf{a}_1 - \mathbf{b}_1| + \cdots + |\mathbf{a_k} - \mathbf{b}_k|$

Theorem (Hirvensalo, M. & Yakaryılmaz)

$\mathsf{AfL}_{\mathbb{Q}} \subseteq \mathsf{L}$

NON-AFFINE LANGUAGES

NON-BAFL LANGUAGES

P a polynomial with nonnegative coefficients and $\deg(P) > 2$,
POLY $= \{a^{P(n)} \mid n \in \mathbb{N}\}$.

PRIME $= \{a^p \mid p \text{ prime}\}$

Theorem (Hirvensalo, M. & Yakaryılmaz, 2017)

POLY $\notin$ BAfL

PRIME $\notin$ BAfL

Proof idea:

Regularity of matrix multiplication cannot express such languages

NON-AFL LANGUAGES

P a polynomial with nonnegative coefficients and $\deg(P) > 2$,
$\mathsf{POLY} = \{a^{P(n)} \mid n \in \mathbb{N}\}$.

$\mathsf{PRIME} = \{a^p \mid p \text{ prime}\}$

Theorem (Hirvensalo, M. & Yakaryılmaz)

$\mathsf{POLY} \notin \mathsf{AfL}_{\mathbb{A}}$

$\mathsf{PRIME} \notin \mathsf{AfL}_{\mathbb{A}}$

Proof idea:

Same but with finer analysis

Non-linear automaton model inspired by quantum ones

→ Can be simulated by logspace Turing machines

→ But cannot recognize all logspace languages

What next ?

→ Finally prove that POLY, PRIME $\notin$ AfL

We only have POLY, PRIME $\notin \mathrm{AfL}_{\mathbb{A}}$

Non-linear automaton model inspired by quantum ones

→ Can be simulated by logspace Turing machines

→ But cannot recognize all logspace languages

What next ?

→ Finally prove that POLY, PRIME $\notin$ AfL

→ Separation between BAfL and AfL?

For stochastic and quantum, separation based on BSL $=$ REG

Non-linear automaton model inspired by quantum ones

→ Can be simulated by logspace Turing machines

→ But cannot recognize all logspace languages

What next ?

→ Finally prove that POLY, PRIME $\notin$ AfL

→ Separation between BAfL and AfL?

→ Fully characterize BAfL and AfL languages

Semi-linear languages ?

Big class for which our non-membership proof fails

Thank you !

HSRM THEME

`HSRM Theme` is released under ›**GNU Public License**‹ : you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation.