Algebraic Automata Theory

Sven Dziadek
Inria Paris

Formal Language Theory

Introduction: Formal Languages

Fix a finite alphabet Σ.

Introduction: Formal Languages

Fix a finite alphabet Σ.

$$\Sigma^* = \bigcup_{n \geq 0} \Sigma^n = \{\sigma_1 \cdots \sigma_n \mid \sigma_i \in \Sigma, n \geq 0\}$$

Example

For $\Sigma = \{a, b\}$:
$\Sigma^* = \{\epsilon, a, b, aa, ab, ba, bb, \ldots\}$ (ϵ is the empty word)

Introduction: Formal Languages

Fix a finite alphabet Σ.

$$\Sigma^* = \bigcup_{n \geq 0} \Sigma^n = \{\sigma_1 \cdots \sigma_n \mid \sigma_i \in \Sigma, n \geq 0\}$$

Example

For $\Sigma = \{a, b\}$:
$\Sigma^* = \{\epsilon, a, b, aa, ab, ba, bb, \ldots\}$ (ϵ is the empty word)

Formal Language

A (formal) language is $L \subseteq \Sigma^*$

Introduction: Regular Expressions

Regular Expressions

Fix a finite alphabet Σ.

–
- $\emptyset$
- ϵ (empty word)
- a (for any $a \in \Sigma$)

are regular expressions

Introduction: Regular Expressions

Regular Expressions

Fix a finite alphabet Σ.

-
  - $\emptyset$
  - ϵ (empty word)
  - a (for any $a \in \Sigma$)

  are regular expressions
- for regular expressions e and f:
  - $e + f$ (alternation)
  - ef (concatenation)
  - e^* (Kleene star)

  are regular expressions

Introduction: Regular Expressions

Regular Expressions

Fix a finite alphabet Σ.

-
  - $\emptyset$
  - ϵ (empty word)
  - a (for any $a \in \Sigma$)

  are regular expressions
- for regular expressions e and f:
  - $e + f$ (alternation)
  - ef (concatenation)
  - e^* (Kleene star)

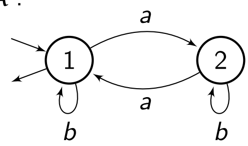
  are regular expressions

Example

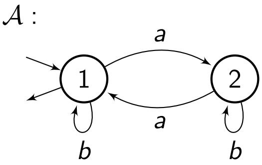
Language of even-length words: $L = ((a+b)(a+b))^*$

Introduction: Finite Automata

$\mathcal{A}$:

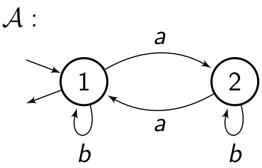


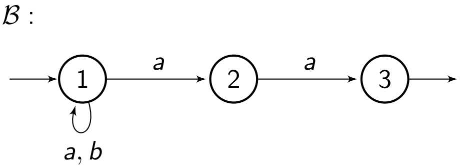
Introduction: Finite Automata



$$\mathcal{L}(\mathcal{A}) = \{w \in \{a,b\} \mid \text{amount of } a \text{ is even}\}$$
$$= b^*(ab^*ab^*)^*$$

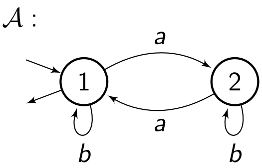
Introduction: Finite Automata

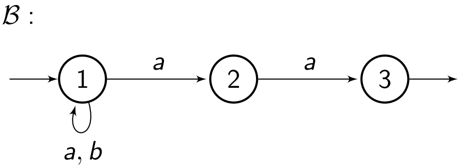




$$\mathcal{L}(\mathcal{A}) = \{w \in \{a, b\} \mid \text{amount of } a \text{ is even}\}$$
$$= b^*(ab^*ab^*)^*$$

Introduction: Finite Automata





$$\mathcal{L}(\mathcal{A}) = \{w \in \{a, b\} \mid \text{amount of } a \text{ is even}\}$$
$$= b^*(ab^*ab^*)^*$$

$$\mathcal{L}(\mathcal{B}) = \{w \in \{a, b\} \mid \text{ends on } aa\}$$
$$= (a + b)^* aa$$

Introduction: Weighted Languages

Consider formal language $L \subseteq \Sigma^*$.

Introduction: Weighted Languages

Consider formal language $L \subseteq \Sigma^*$.

Another point of view: $L \colon \Sigma^* \to \{\text{True}, \text{False}\}$

Introduction: Weighted Languages

Consider formal language $L \subseteq \Sigma^*$.

Another point of view: $L\colon \Sigma^* \to \underbrace{\{\text{True}, \text{False}\}}_{\text{set } S}$

Generalize:

Introduction: Weighted Languages

Consider formal language $L \subseteq \Sigma^*$.

Another point of view: $L\colon \Sigma^* \to \underbrace{\{\text{True}, \text{False}\}}_{\text{set } S}$

Generalize:

$$s\colon \Sigma^* \to S \qquad \text{(weighted languages, called } \textit{series}\text{)}$$

Introduction: Weighted Languages

Consider formal language $L \subseteq \Sigma^*$.

Another point of view: $L \colon \Sigma^* \to \underbrace{\{\text{True}, \text{False}\}}_{\text{set } S}$

Generalize:

$$s \colon \Sigma^* \to S \qquad \text{(weighted languages, called \textit{series})}$$

$$s \colon \Sigma^\omega \to S \qquad \text{(ω-\textit{series})}$$

Introduction: Weighted Languages

Consider formal language $L \subseteq \Sigma^*$.

Another point of view: $L\colon \Sigma^* \to \underbrace{\{\text{True}, \text{False}\}}_{\text{set } S}$

Generalize:

$$s\colon \Sigma^* \to S \qquad \text{(weighted languages, called } \textit{series}\text{)}$$

$$s\colon \Sigma^\omega \to S \qquad (\omega\text{-}\textit{series})$$

Examples

- counting
- optimization (costs or gains)
- probabilities
- transducer
- average, discounting

Introduction: Weighted Automata (on Semirings)

$$\|\mathcal{A}\| : \Sigma^* \to S$$

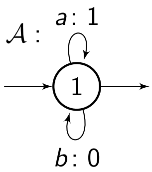
Introduction: Weighted Automata (on Semirings)

$S = (\mathbb{N}^{\infty}, \min, +, \infty, 0)$

$$\|\mathcal{A}\| : \Sigma^* \to S$$

Introduction: Weighted Automata (on Semirings)

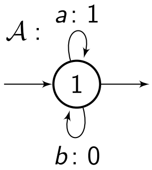
$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$



$\|\mathcal{A}\| : \Sigma^* \to S$

Introduction: Weighted Automata (on Semirings)

$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$



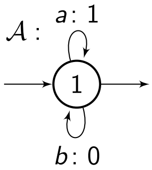
$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a$

Introduction: Weighted Automata (on Semirings)

$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$

$S = (\mathbb{N}, +, \cdot, 0, 1)$



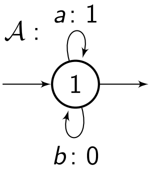
$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a$

Introduction: Weighted Automata (on Semirings)

$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$

$\mathcal{A}$:

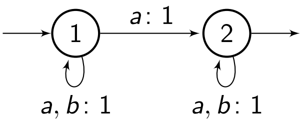


$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a$

$S = (\mathbb{N}, +, \cdot, 0, 1)$

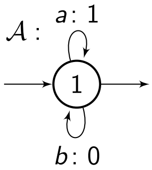
$\mathcal{B}$:



Introduction: Weighted Automata (on Semirings)

$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$

$\mathcal{A}$:

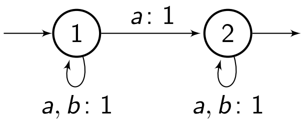


$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a$

$S = (\mathbb{N}, +, \cdot, 0, 1)$

$\mathcal{B}$:

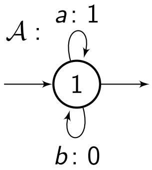


Example

$a \quad b \quad a \quad b \quad a$

Introduction: Weighted Automata (on Semirings)

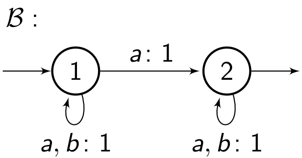
$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$



$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a$

$S = (\mathbb{N}, +, \cdot, 0, 1)$

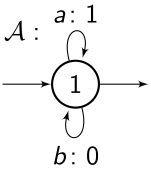


Example

$$
\begin{array}{cccccc}
 & a & b & a & b & a \\
1 & \to 1 & \to 1 & \to 1 & \to 1 & \to 2 \\
1 & \to 1 & \to 1 & \to 2 & \to 2 & \to 2 \\
1 & \to 2 & \to 2 & \to 2 & \to 2 & \to 2
\end{array}
$$

Introduction: Weighted Automata (on Semirings)

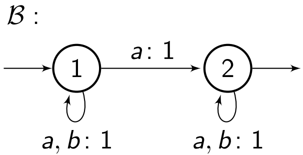
$S = (\mathbb{N}^{\infty}, \min, +, \infty, 0)$



$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a$

$S = (\mathbb{N}, +, \cdot, 0, 1)$



Example

$a \quad b \quad a \quad b \quad a$

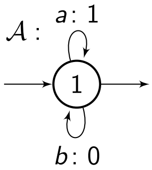
$1 \to 1 \to 1 \to 1 \to 1 \to 2$ wt 1

$1 \to 1 \to 1 \to 2 \to 2 \to 2$ wt 1

$1 \to 2 \to 2 \to 2 \to 2 \to 2$ wt 1

Introduction: Weighted Automata (on Semirings)

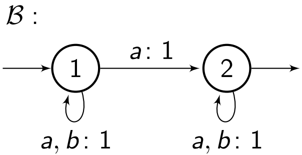
$S = (\mathbb{N}^{\infty}, \min, +, \infty, 0)$



$\|\mathcal{A}\| \colon \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a$

$S = (\mathbb{N}, +, \cdot, 0, 1)$



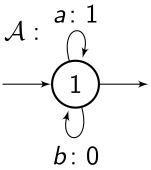
Example

a	b	a	b	a		
$1 \to 1$	$\to 1$	$\to 1$	$\to 1$	$\to 2$		wt 1
$1 \to 1$	$\to 1$	$\to 2$	$\to 2$	$\to 2$		wt 1
$1 \to 2$	$\to 2$	$\to 2$	$\to 2$	$\to 2$		wt 1
						sum: 3

Introduction: Weighted Automata (on Semirings)

$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$

$\mathcal{A}$:

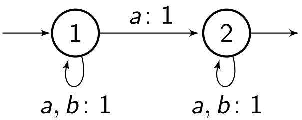


$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a = \|\mathcal{B}\|(w)$

$S = (\mathbb{N}, +, \cdot, 0, 1)$

$\mathcal{B}$:

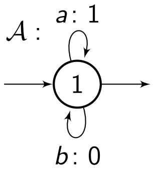


Example

a	b	a	b	a	
$1 \to 1$	$\to 1$	$\to 1$	$\to 1$	$\to 2$	wt 1
$1 \to 1$	$\to 1$	$\to 2$	$\to 2$	$\to 2$	wt 1
$1 \to 2$	$\to 2$	$\to 2$	$\to 2$	$\to 2$	wt 1
					sum: 3

Introduction: Weighted Automata (on Semirings)

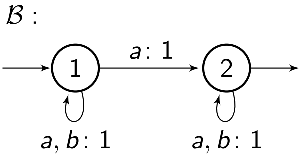
$S = (\mathbb{N}^\infty, \min, +, \infty, 0)$



$\|\mathcal{A}\| : \Sigma^* \to S$

$\|\mathcal{A}\|(w) = |w|_a = \|\mathcal{B}\|(w)$

$S = (\mathbb{N}, +, \cdot, 0, 1)$



Example

a	b	a	b	a		
1 →1	→1	→1	→1	→2		wt 1
1 →1	→1	→2	→2	→2		wt 1
1 →2	→2	→2	→2	→2		wt 1
					sum:	3

Examples

- unweighted (Boolean semiring): $(\mathbb{B}, \vee, \wedge, \bot, \top)$
- probabilities: $(\mathbb{Q}_+, +, \cdot, 0, 1)$
- transducer: $(2^{\Sigma^*}, \cup, \cdot, \emptyset, \{\epsilon\})$
- Viterbi: $([0, 1], \max, \cdot, 0, 1)$

Conway Semirings

Identities for Conway Semirings

Conway semirings are star-semirings with:

- sum-star-equation: $(a+b)^* = (a^*b)^*a^*$
- product-star-equation: $(ab)^* = 1 + a(ba)^*b$
- it follows: $a^* = 1 + aa^*$ and $(ab)^*a = a(ba)^*$

Identities for Conway Semirings

Conway semirings are star-semirings with:

- sum-star-equation: $(a+b)^* = (a^*b)^*a^*$
- product-star-equation: $(ab)^* = 1 + a(ba)^*b$
- it follows: $a^* = 1 + aa^*$ and $(ab)^*a = a(ba)^*$

Extension to infinite words (star-omega-semirings):

- sum-omega equation: $(a+b)^\omega = (a^*b)^\omega + (a^*b)^*a^\omega$
- product-omega equation: $(ab)^\omega = a(ba)^\omega$
- it follows: $aa^\omega = a^\omega$

Formalization in a Proof Assistant: Baby Steps in Cubical Agda

```
record IsConwaySemiring {R : Type}
                  (0r 1r : R) (_+_ _·_ : R → R → R) (_* : R → R) : Type where
  field
    cIsSemiring   : IsSemiring 0r 1r _+_ _·_
    sumStar       : (x y : R) → ((x + y)*) = (((x *) · y)*) · (x *)
    productStar   : (x y : R) → ((x · y)*) = 1r + ((x · ((y · x)*)) · y)
```

Example

```
starIdentityPlusL : (a : fst S) → a * = 1r + a · (a *)
starIdentityPlusL a =
  a *                        =< ap (\ x → x *) (sym (IsConwaySemiring.·IdR ics a)) >
  (a · 1r)*                  =< productStar a 1r >
  1r + a · (1r · a)* · 1r =< ap (\ x → 1r + x) (IsConwaySemiring.·IdR ics (a · (1r
  1r + a · (1r · a)*         =< ap (\ x → 1r + a · (x *)) (IsConwaySemiring.·IdL ics
  1r + a · a *               \qed
```

Related Work

Damien Pous:
Kleene Algebra with Tests in Coq:

Related Work

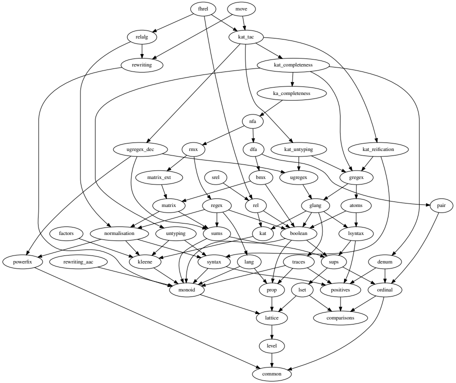
Damien Pous:

Kleene Algebra with Tests in Coq:

Kleene Algebra: Similar to the above but idempotent

Related Work

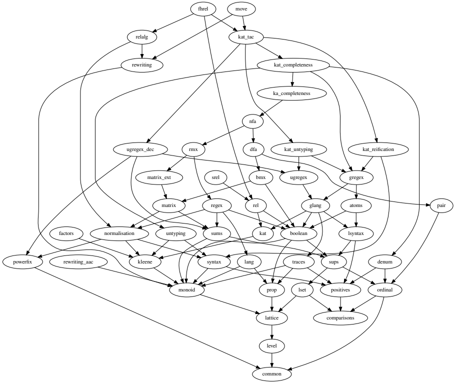
Damien Pous:
Kleene Algebra with Tests in Coq:



Kleene Algebra: Similar to the above but idempotent

Related Work

Damien Pous:
Kleene Algebra with Tests in Coq:



Georg Struth:
Kleene Algebra in Isabelle/HOL:

- More than 1000 lemmas

Kleene Algebra: Similar to the above but idempotent

Weighted Automata

with matrices

(In)Finite Applications of a Matrix

Transition Matrix

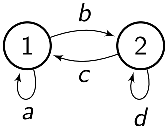
Intuition: *adjacency matrix* of a finite automaton

(In)Finite Applications of a Matrix

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

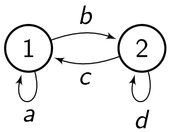


(In)Finite Applications of a Matrix

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$



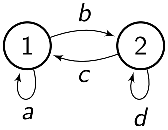
$$M^* = \begin{pmatrix} (a + bd^*c)^* & (a + bd^*c)^*bd^* \\ (d + ca^*b)^*ca^* & (d + ca^*b)^* \end{pmatrix}$$

(In)Finite Applications of a Matrix

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$



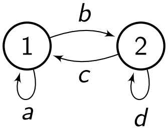
$$M^* = \begin{pmatrix} (a + bd^* c)^* & (a + bd^* c)^* bd^* \\ (d + ca^* b)^* ca^* & (d + ca^* b)^* \end{pmatrix}$$

(In)Finite Applications of a Matrix

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$



$$M^* = \begin{pmatrix} (a + bd^* c)^* & (a + bd^* c)^* bd^* \\ (d + ca^* b)^* ca^* & (d + ca^* b)^* \end{pmatrix}$$

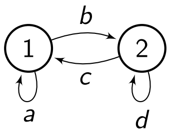
$$M^\omega = \begin{pmatrix} (a + bd^* c)^\omega + (a + bd^* c)^* bd^\omega \\ (d + ca^* b)^* ca^\omega + (d + ca^* b)^\omega \end{pmatrix}$$

(In)Finite Applications of a Matrix

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$



$$M^* = \begin{pmatrix} (a + bd^*c)^* & (a + bd^*c)^*bd^* \\ (d + ca^*b)^*ca^* & (d + ca^*b)^* \end{pmatrix}$$

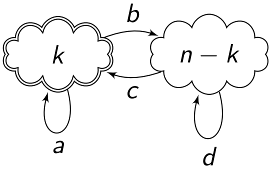
$$M^\omega = \begin{pmatrix} (a + bd^*c)^\omega + (a + bd^*c)^*bd^\omega \\ (d + ca^*b)^*ca^\omega + (d + ca^*b)^\omega \end{pmatrix}$$

(In)Finite Applications of a Matrix - Büchi Condition

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \underbrace{\begin{pmatrix} a \\ c \end{pmatrix}}_{k}\underbrace{\begin{pmatrix} b \\ d \end{pmatrix}}_{n-k} \quad M = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix} \}k \\ \}n-k \end{matrix}$$



$$M^* = \begin{pmatrix} (a+bd^*c)^* & (a+bd^*c)^*bd^* \\ (d+ca^*b)^*ca^* & (d+ca^*b)^* \end{pmatrix}$$

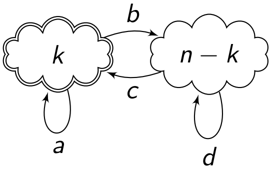
$$M^\omega = \begin{pmatrix} (a+bd^*c)^\omega + (a+bd^*c)^*bd^\omega \\ (d+ca^*b)^*ca^\omega + (d+ca^*b)^\omega \end{pmatrix}$$

(In)Finite Applications of a Matrix - Büchi Condition

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \underbrace{\begin{pmatrix} a \\ c \end{pmatrix}}_{k}\underbrace{\begin{pmatrix} b \\ d \end{pmatrix}}_{n-k} \quad \begin{matrix} \}k \\ \}n-k \end{matrix}$$



$$M^* = \begin{pmatrix} (a + bd^*c)^* & (a + bd^*c)^* bd^* \\ (d + ca^*b)^* ca^* & (d + ca^*b)^* \end{pmatrix}$$

$$M^\omega = \begin{pmatrix} (a + bd^*c)^\omega + (a + bd^*c)^* bd^\omega \\ (d + ca^*b)^* ca^\omega + (d + ca^*b)^\omega \end{pmatrix}$$

$$M^{\omega,k} = \begin{pmatrix} (a + bd^*c)^\omega \\ d^*c(a + bd^*c)^\omega \end{pmatrix}$$

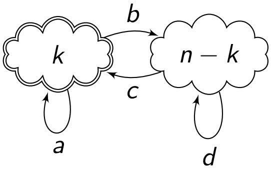
(In)Finite Applications of a Matrix - Büchi Condition

Transition Matrix

Intuition: *adjacency matrix* of a finite automaton

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix} \}k \\ \}n-k \end{matrix}$$

$$\underbrace{\quad}_{k} \quad \underbrace{\quad}_{n-k}$$



$$M^* = \begin{pmatrix} (a+bd^*c)^* & (a+bd^*c)^*bd^* \\ (d+ca^*b)^*ca^* & (d+ca^*b)^* \end{pmatrix}$$

$$M^\omega = \begin{pmatrix} (a+bd^*c)^\omega + (a+bd^*c)^*bd^\omega \\ (d+ca^*b)^*ca^\omega + (d+ca^*b)^\omega \end{pmatrix}$$

$$M^{\omega,k} = \begin{pmatrix} (a+bd^*c)^\omega \\ d^*c(a+bd^*c)^\omega \end{pmatrix}$$

Formalization in a Proof Assistant (2)

Isabell/HOL complicated with matrices?

because no dependent types

Formalization in a Proof Assistant (2)

Isabell/HOL complicated with matrices?

because no dependent types

I am fascinated by Homotopy Type Theory (Cubical Agda)

but should I bother?

Conclusion

Algebraic Automata Theory

– Extend automata (and also grammars) to matrices

Conclusion

Algebraic Automata Theory

- Extend automata (and also grammars) to matrices

Open Problems

- Generalize/extend more unweighted results
- Completeness
- Formalization in proof assistant

Conclusion

Algebraic Automata Theory

- Extend automata (and also grammars) to matrices

Open Problems

- Generalize/extend more unweighted results
- Completeness
- Formalization in proof assistant

Thank you for your attention!