Cyclic Proofs and jumping automata

Denis Kuperberg Laureline Pinault Damien Pous

LIP, ENS Lyon

Highlights of Logic, Games and Automata
Warsaw

Friday 20th 2019

Cyclic proofs

Regular expressions

$$e, f := 1 \mid a \in A \mid e \cdot f \mid e + f \mid e^*$$

Context: Cyclic proofs for inclusion of expressions [Das, Pous '17]

- Infinite proof trees, with root of the form $e \vdash f$.

$$\dfrac{\dfrac{\dfrac{}{1 \vdash 1}\,(\mathrm{Ax})}{1 \vdash a^*} \qquad \dfrac{\dfrac{}{a \vdash a}\,(\mathrm{Ax}) \qquad a^* \vdash a^*}{a, a^* \vdash a^*}}{a^* \vdash a^*}$$

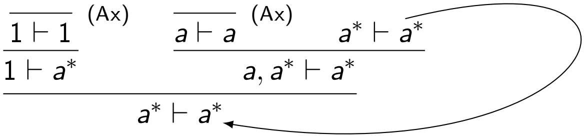
Cyclic proofs

Regular expressions

$$e, f := 1 \mid a \in A \mid e \cdot f \mid e + f \mid e^*$$

Context: Cyclic proofs for inclusion of expressions [Das, Pous '17]

- Infinite proof trees, with root of the form $e \vdash f$.



- Validity condition on infinite branches

Cyclic proofs

Regular expressions

$$e, f := 1 \mid a \in A \mid e \cdot f \mid e + f \mid e^*$$

<u>Context:</u> Cyclic proofs for inclusion of expressions [Das, Pous '17]

- Infinite proof trees, with root of the form $e \vdash f$.

$$\dfrac{\dfrac{}{1 \vdash 1}\,(\mathsf{Ax}) \qquad \dfrac{\dfrac{}{a \vdash a}\,(\mathsf{Ax}) \qquad a^* \vdash a^*}{a, a^* \vdash a^*}}{a^* \vdash a^*}$$

- Validity condition on infinite branches
- $\exists$ proof of $e \vdash f \Leftrightarrow L(e) \subseteq L(f)$.

Computational interpretation

Proof of $e \vdash f$

$\rightsquigarrow$ Computational interpretation

Program with input from e and output in f.

Several proofs of the same statement
$\Leftrightarrow$ Several programs of the same type

Example: $a \vdash a + a$

Computational interpretation

Proof of $e \vdash f$

$\rightsquigarrow$ Computational interpretation

Program with input from e and output in f.

Several proofs of the same statement
$\Leftrightarrow$ Several programs of the same type

Example: $a \vdash a + a$

Curry-Howard isomorphism, typed programming,...
Well-understood for finite proofs, active field for infinite proofs.

This work: Computational content of proofs for regular expressions.

- Boolean type $2 = 1 + 1$

This work: Computational content of proofs for regular expressions.

- Boolean type $2 = 1 + 1$
- Add *structural* rules corresponding to simple natural programs

This work: Computational content of proofs for regular expressions.

- Boolean type $2 = 1 + 1$
- Add *structural* rules corresponding to simple natural programs
- Study the expressive power of regular proofs (finite graphs)

This work: Computational content of proofs for regular expressions.

- Boolean type $2 = 1 + 1$
- Add *structural* rules corresponding to simple natural programs
- Study the expressive power of regular proofs (finite graphs)
- Focus on proofs for languages:

$$\text{Proof } \pi \text{ of } A^* \vdash 2 \rightsquigarrow \text{Language } L(\pi) \subseteq A^*$$

Proof system

Expressions $e := A \mid A^*$
Sequents $E, F = e_1, e_2, \ldots, e_n$
Proof system with extra rules for basic data manipulation:

$$\frac{}{\vdash 2}\ (\mathbf{tt}) \qquad \frac{}{\vdash 2}\ (\mathbf{ff})$$

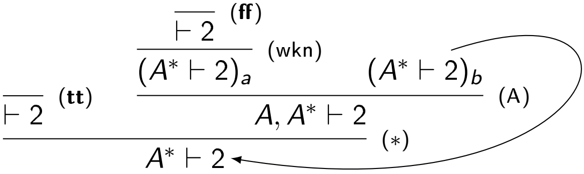
$$\frac{E, F \vdash 2}{E, e, F \vdash 2}\ (\mathsf{wkn}) \qquad \frac{E, e, e, F \vdash 2}{E, e, F \vdash 2}\ (\mathsf{ctr})$$

$$\frac{(E, F \vdash 2)_{a \in A}}{E, A, F \vdash 2}\ (\mathsf{A}) \qquad \frac{E, F \qquad E, A, A^*, F \vdash 2}{E, A^*, F \vdash 2}\ (*)$$

Proofs as language acceptors

What are the languages computed by cyclic proofs ?

Example on alphabet $\{a, b\}$:

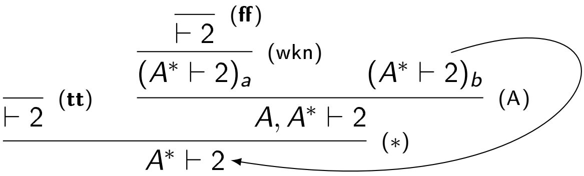


Language recognized: b^*.

Proofs as language acceptors

What are the languages computed by cyclic proofs ?

Example on alphabet $\{a, b\}$:



Language recognized: b^*.

Lemma

Without contraction, the system captures exactly regular languages.

With contractions

With contractions: what class of language ?

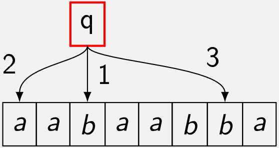
With contractions

<u>With contractions</u>: what class of language ?

Jumping Multihead Automata

A JMA is an automaton with k reading heads.

$$\textit{Transitions: } Q \times A^k \to Q \times \{\blacktriangleright\!|, ⁘, J_1, \ldots, J_k\}^k$$



- $\blacktriangleright\!|$: advance one step
- ⁘: stay in place
- J_i: jump to the position of head i

Expressive power of JMA

Theorem

Cyclic proofs and JMA recognize the same class of languages.

Example: $\{a^{2^n} \mid n \in \mathbb{N}\}$ is accepted by a 2-head JMA.

Expressive power of JMA

Theorem

Cyclic proofs and JMA recognize the same class of languages.

Example: $\{a^{2^n} \mid n \in \mathbb{N}\}$ is accepted by a 2-head JMA.

Comparison with Multihead Automata:
[Holzer, Kutrib, Malcher 2008]

1-way Multihead $\subseteq$ JMA $\subseteq$ 2-way Multihead

Expressive power of JMA

Theorem

Cyclic proofs and JMA recognize the same class of languages.

Example: $\{a^{2^n} \mid n \in \mathbb{N}\}$ is accepted by a 2-head JMA.

Comparison with Multihead Automata:
[Holzer, Kutrib, Malcher 2008]

1-way Multihead	$\subseteq$	JMA	$\subseteq$	2-way Multihead
Emptiness Undecidable				=DLogSpace

What next ?

- Add the cut rule
- Corresponds to composition of functions
- Sequents $1^* \vdash 1^*$: functions $\mathbb{N} \to \mathbb{N}$

Work in progress:

No contraction	=	Primitive Recursive
With contraction	=	System T

What next ?

- Add the cut rule
- Corresponds to composition of functions
- Sequents $1^* \vdash 1^*$: functions $\mathbb{N} \to \mathbb{N}$

Work in progress:

No contraction	=	Primitive Recursive
With contraction	=	System T

Thank you for your attention !