Cyclic system for an algebraic theory of alternating parity automata

Anupam Das Abhishek De

University of Birmingham

34th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods

Reykjavík, Iceland,
28th September 2025.

Outline

1. Languages of infinite words
2. A cyclic proof system
3. Metalogical results
4. Conclusions

An ω-*automaton* runs on infinite words (of length $\leqslant \omega$).

An ω-*automaton* runs on infinite words (of length $\leqslant \omega$).

Alternating Parity Automata (APAs) are ω-automata with:

- States may be $\exists$xistentially or $\forall$niversally branching.
- Each state is coloured by a natural number.
- An run is **accepting** if the least colour occurring infinitely often is even.

Formally the dynamics of runs and definition of word acceptance are given game theoretically, - more on this later.

An ω-*automaton* runs on infinite words (of length $\leqslant \omega$).

Alternating Parity Automata (APAs) are ω-automata with:

- States may be $\exists$xistentially or $\forall$niversally branching.
- Each state is coloured by a natural number.
- An run is **accepting** if the least colour occurring infinitely often is even.

Formally the dynamics of runs and definition of word acceptance are given game theoretically, - more on this later.

Fact

$L \subseteq \mathcal{A}^\omega$ is recognised by an APA $\iff L = \bigcup_{i<n} A_i B_i^\omega$, for A_i, B_i regular and $\not\ni \varepsilon$.

*Call such a language ω-**regular**.*

An ω-*automaton* runs on infinite words (of length $\leqslant \omega$).

Alternating Parity Automata (APAs) are ω-automata with:

- States may be $\exists$xistentially or $\forall$niversally branching.
- Each state is coloured by a natural number.
- An run is **accepting** if the least colour occurring infinitely often is even.

Formally the dynamics of runs and definition of word acceptance are given game theoretically, - more on this later.

Fact

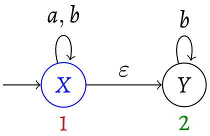
$L \subseteq \mathcal{A}^\omega$ is recognised by an APA $\iff L = \bigcup_{i<n} A_i B_i^\omega$, for A_i, B_i regular and $\not\ni \varepsilon$.

*Call such a language ω-**regular**.*

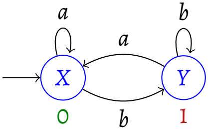
NB: there are many equivalent models of automata for ω-regular languages, but APAs enjoy elegant symmetries.

Examples of APAs

finitely many as :



infinitely many as :

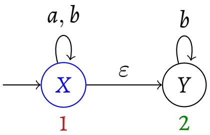


Key

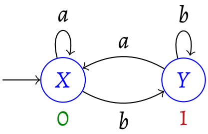
: existential state
: universal state
n : even colour
n : odd colour

EXAMPLES OF APAS

finitely many as :



infinitely many as :



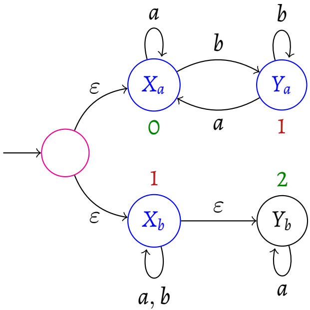
Key

: existential state
: universal state
n : even colour
n : odd colour

infinitely many as

and :

finitely many bs



We can reason about APAs via expressions with fixed points.

RLL expressions:

$$
\begin{array}{rcccccccccc}
e, f, \ldots & ::= & X & \mid & ae & \mid & 0 & \mid & e + f & \mid & \mu X\, e \\
 & & & & & \mid & \top & \mid & e \cap f & \mid & \nu X\, e
\end{array}
$$

NB: symmetry of APAs is reflected by duality of syntax.

This syntax can be seen as an *alternative* to the modal logic μLTL.

We can reason about APAs via expressions with fixed points.

RLL expressions:

$$
\begin{array}{rclcccccccccc}
e, f, \dots & ::= & X & \mid & ae & \mid & 0 & \mid & e + f & \mid & \mu X\, e \\
& & & & & \mid & \top & \mid & e \cap f & \mid & \nu X\, e
\end{array}
$$

NB: symmetry of APAs is reflected by duality of syntax.

This syntax can be seen as an *alternative* to the modal logic $\mu\mathsf{LTL}$.

Language semantics:

$$
\begin{aligned}
\mathcal{L}(A) &:= A & \qquad \mathcal{L}(ae) &:= \{aw \mid w \in \mathcal{L}(e)\} \\
\mathcal{L}(0) &:= \varnothing & \mathcal{L}(\top) &:= \mathcal{A}^\omega \\
\mathcal{L}(e+f) &:= \mathcal{L}(e) \cup \mathcal{L}(f) & \mathcal{L}(e \cap f) &:= \mathcal{L}(e) \cap \mathcal{L}(f) \\
\mathcal{L}(\mu X e(X)) &:= \mathrm{LFP}[A \mapsto \mathcal{L}(e(A))] & \mathcal{L}(\nu X e(X)) &:= \mathrm{GFP}[A \mapsto \mathcal{L}(e(A))]
\end{aligned}
$$

We can reason about APAs via expressions with fixed points.

RLL expressions:

$$
\begin{array}{ccccccccccc}
e, f, \dots & ::= & X & \mid & ae & \mid & 0 & \mid & e+f & \mid & \mu X\, e \\
 & & & & & \mid & \top & \mid & e \cap f & \mid & \nu X\, e
\end{array}
$$

NB: symmetry of APAs is reflected by duality of syntax.

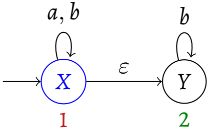
This syntax can be seen as an *alternative* to the modal logic μLTL.

Language semantics:

$$
\begin{aligned}
\mathcal{L}(A) &:= A & \mathcal{L}(ae) &:= \{aw \mid w \in \mathcal{L}(e)\} \\
\mathcal{L}(0) &:= \varnothing & \mathcal{L}(\top) &:= \mathcal{A}^\omega \\
\mathcal{L}(e+f) &:= \mathcal{L}(e) \cup \mathcal{L}(f) & \mathcal{L}(e \cap f) &:= \mathcal{L}(e) \cap \mathcal{L}(f) \\
\mathcal{L}(\mu X e(X)) &:= \mathrm{LFP}[A \mapsto \mathcal{L}(e(A))] & \mathcal{L}(\nu X e(X)) &:= \mathrm{GFP}[A \mapsto \mathcal{L}(e(A))]
\end{aligned}
$$

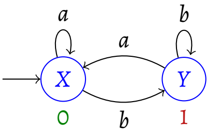
NB: *Knaster-Tarski Theorem* $\implies$ $\mathcal{L}(e)$ is well-defined.

finitely many as :



$f_a : \mu X(aX + bX + \nu Y(bY))$

infinitely many as :



$i_a : \nu X \mu Y(aX + bY)$

Key

○ : existential state

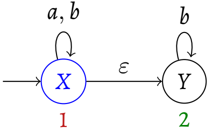
○ : universal state

n : even colour

n : odd colour

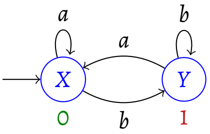
Examples of RLL expressions

finitely many as :



$f_a : \mu X(aX+bX+\nu Y(bY))$

infinitely many as :



$i_a : \nu X \mu Y(aX+bY)$

Key

: existential state

: universal state

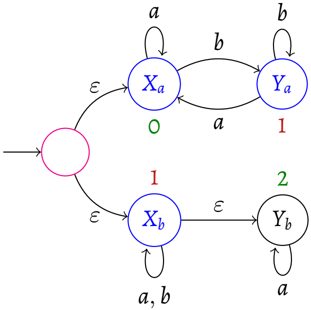
n : even colour

n : odd colour

infinitely many as

and :

finitely many bs

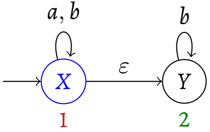


$\nu X_a \mu Y_a \, (aX_a+bY_a)$

$\cap$

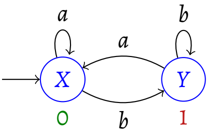
$\mu X_b(aX_b+bX_b+\nu Y_b(bY_b))$

finitely many as :



$f_a : \mu X(aX+bX+\nu Y(bY))$

infinitely many as :



$i_a : \nu X \mu Y(aX+bY)$

Key

: existential state

: universal state

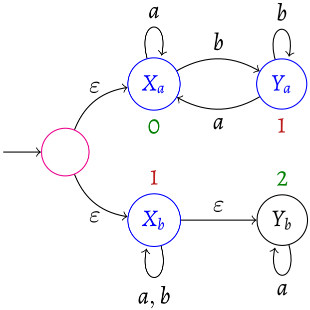
n : even colour

n : odd colour

infinitely many as

and :

finitely many bs



$\nu X_a \mu Y_a \, (aX_a+bY_a)$

$\cap$

$\mu X_b(aX_b+bX_b+\nu Y_b(bY_b))$

NB: *equivalence* between an expression and associated APA is not immediate...

Positions: pairs (w, e) where $w \in \mathcal{A}^\omega$ and e an expression.

Players: *Eloise* ($\exists$) and *Abelard* ($\forall$).

Positions: pairs (w, e) where $w \in \mathcal{A}^\omega$ and e an expression.

Players: *Eloise* ($\exists$) and *Abelard* ($\forall$).

Position	Player	Available moves
(aw, ae)	-	(w, e)
(aw, be) with $a \neq b$	$\exists$	
$(w, 0)$	$\exists$	
$(w, \top)$	$\forall$	
$(w, e + f)$	$\exists$	(w, e), (w, f)
$(w, e \cap f)$	$\forall$	(w, e), (w, f)
$(w, \mu X e(X))$	-	$(w, e(\mu X e(X)))$
$(w, \nu X e(X))$	-	$(w, e(\nu X e(X)))$

Positions: pairs (w, e) where $w \in \mathcal{A}^\omega$ and e an expression.

Players: *Eloise* ($\exists$) and *Abelard* ($\forall$).

Position	Player	Available moves
(aw, ae)	-	(w, e)
(aw, be) with $a \neq b$	$\exists$	
$(w, 0)$	$\exists$	
$(w, \top)$	$\forall$	
$(w, e + f)$	$\exists$	$(w, e), (w, f)$
$(w, e \cap f)$	$\forall$	$(w, e), (w, f)$
$(w, \mu X e(X))$	-	$(w, e(\mu X e(X)))$
$(w, \nu X e(X))$	-	$(w, e(\nu X e(X)))$

$\exists$ **wins** an infinite play if smallest infinitely occurring expression is a ν-expression.

Positions: pairs (w, e) where $w \in \mathcal{A}^\omega$ and e an expression.

Players: *Eloise* ($\exists$) and *Abelard* ($\forall$).

Position	Player	Available moves
(aw, ae)	-	(w, e)
(aw, be) with $a \neq b$	$\exists$	
$(w, 0)$	$\exists$	
$(w, \top)$	$\forall$	
$(w, e + f)$	$\exists$	$(w, e), (w, f)$
$(w, e \cap f)$	$\forall$	$(w, e), (w, f)$
$(w, \mu Xe(X))$	-	$(w, e(\mu Xe(X)))$
$(w, \nu Xe(X))$	-	$(w, e(\nu Xe(X)))$

$\exists$ **wins** an infinite play if smallest infinitely occurring expression is a ν-expression.

By now standard techniques we can show:

Theorem (Adequacy)

$w \in \mathcal{L}(e) \iff$ *$\exists$ has a (positional) winning strategy from* (w, e).

$\rightsquigarrow$ an RLL expression and its associated APA compute the same ω-language.

Algebraic vs modal perspectives:

There is already a syntax for APAs: the modal logic $\mu\mathsf{LTL}$!

- RLL expressions are based in the language of lattices while $\mu\mathsf{LTL}$ is based in the language of Boolean algebras. (*no complements!*)
- Resulting theories are subtly but formally different: there are *Right-linear Lattices* that are not Boolean algebras, and not complete lattices [Kai95, DD25].
- Nonetheless, we are heavily inspired by previous work for μ-calculi, in particular [DHL06]: we are using now standard techniques, cf. [NW96].

Algebraic vs modal perspectives:

There is already a syntax for APAs: the modal logic μLTL!

- RLL expressions are based in the language of lattices while μLTL is based in the language of Boolean algebras. (*no complements!*)
- Resulting theories are subtly but formally different: there are *Right-linear Lattices* that are not Boolean algebras, and not complete lattices [Kai95, DD25].
- Nonetheless, we are heavily inspired by previous work for μ-calculi, in particular [DHL06]: we are using now standard techniques, cf. [NW96].

The algebraic context:

Our work is building on the *algebraic tradition*:

- *Kleene algebras*, for regular expressions, recast as *Right-linear algebras* (RLAs) [DD24b]. (*no multiplication!*) Both may be construed a theory of NFAs.
- RLLs strictly extend RLAs, unlike ω*-regular algebras*.
  Theory of APAs vs theory of Büchi automata.

Algebraic vs modal perspectives:

There is already a syntax for APAs: the modal logic μLTL!

- RLL expressions are based in the language of lattices while μLTL is based in the language of Boolean algebras. (*no complements!*)
- Resulting theories are subtly but formally different: there are *Right-linear Lattices* that are not Boolean algebras, and not complete lattices [Kai95, DD25].
- Nonetheless, we are heavily inspired by previous work for μ-calculi, in particular [DHL06]: we are using now standard techniques, cf. [NW96].

The algebraic context:

Our work is building on the *algebraic tradition*:

- *Kleene algebras*, for regular expressions, recast as *Right-linear algebras* (RLAs) [DD24b]. (*no multiplication!*) Both may be construed a theory of NFAs.
- RLLs strictly extend RLAs, unlike ω*-regular algebras*.
  Theory of APAs vs theory of Büchi automata.

This work is an application of cyclic proofs to algebras for regularity.

Outline

1. $(0, \top, +, \cap)$ forms a *bounded distributive lattice*.
2. Each $a \in \mathcal{A}$ is a (lower) semibounded lattice homomorphism:

$$
\begin{aligned}
a0 &= 0 \\
a(e+f) &= ae + af \\
a(e \cap f) &= ae \cap af
\end{aligned}
$$

1. $(0, \top, +, \cap)$ forms a *bounded distributive lattice*.
2. Each $a \in \mathcal{A}$ is a (lower) semibounded lattice homomorphism:

$$a0 = 0$$
$$a(e + f) = ae + af$$
$$a(e \cap f) = ae \cap af$$

3. The ranges of $a \in \mathcal{A}$ partition $\mathcal{P}(\mathcal{A}^\omega)$:

$$ae \cap bf = 0 \qquad \text{for } a \neq b$$
$$\top = \sum_{a \in \mathcal{A}} a\top$$

1. $(0, \top, +, \cap)$ forms a *bounded distributive lattice*.
2. Each $a \in \mathcal{A}$ is a (lower) semibounded lattice homomorphism:

$$\begin{aligned} a0 &= 0 \\ a(e+f) &= ae + af \\ a(e \cap f) &= ae \cap af \end{aligned}$$

3. The ranges of $a \in \mathcal{A}$ partition $\mathcal{P}(\mathcal{A}^\omega)$:

$$\begin{aligned} ae \cap bf &= 0 && \text{for } a \neq b \\ \top &= \sum_{a \in \mathcal{A}} a\top \end{aligned}$$

4. $\mu X e(X)$ is a least prefixed point of $X \mapsto e(X)$:

$$\begin{aligned} &e(\mu X e(X)) \leqslant \mu X e(X) \\ &e(f) \leqslant f \implies \mu X e(X) \leqslant f \end{aligned}$$

5. $\nu X e(X)$ is a greatest postfixed point of $X \mapsto e(X)$:

$$\begin{aligned} &\nu X e(X) \leqslant e(\nu X e(X)) \\ &f \leqslant e(f) \implies f \leqslant \nu X e(X) \end{aligned}$$

Sequents: $\Gamma \to \Delta$, where Γ, Δ are *sets* of formulas.

Starting point: a standard system for bounded distributive lattices.

Sequents: $\Gamma \to \Delta$, where Γ, Δ are *sets* of formulas.

Starting point: a standard system for bounded distributive lattices.

Definition

The system $\widehat{\mathsf{RLL}}$ has the following additional rules:

$\mathcal{A}$ **rules:**

$$\mathsf{p}\text{-}l\,\frac{}{ae, bf \to}\; a \neq b \qquad \mathsf{h}_a\,\frac{\Gamma \to \Delta}{a\Gamma \to a\Delta}\;\Gamma \neq \varnothing \qquad \mathsf{p}\text{-}r\,\frac{\{\to \Gamma_a\}_{a\in\mathcal{A}}}{\to \{a\Gamma_a\}_{a\in\mathcal{A}}}$$

Fixed point rules:

$$\mu\text{-}l\,\frac{\Gamma, e(\mu X e(X)) \to \Delta}{\Gamma, \mu X e(X) \to \Delta} \qquad \mu\text{-}r\,\frac{\Gamma \to \Delta, e(\mu X e(X))}{\Gamma \to \Delta, \mu X e(X)}$$

$$\nu\text{-}l\,\frac{\Gamma, e(\nu X e(X)) \to \Delta}{\Gamma, \nu X e(X) \to \Delta} \qquad \nu\text{-}r\,\frac{\Gamma \to \Delta, e(\nu X e(X))}{\Gamma \to \Delta, \nu X e(X)}$$

Sequents: $\Gamma \to \Delta$, where Γ, Δ are *sets* of formulas.

Starting point: a standard system for bounded distributive lattices.

Definition

The system $\widehat{\mathsf{RLL}}$ has the following additional rules:

$\mathcal{A}$ **rules:**

$$\mathsf{p}\text{-}l\ \frac{}{ae, bf \to}\ a \neq b \qquad \mathsf{h}_a\ \frac{\Gamma \to \Delta}{a\Gamma \to a\Delta}\ \Gamma \neq \varnothing \qquad \mathsf{p}\text{-}r\ \frac{\{\to \Gamma_a\}_{a \in \mathcal{A}}}{\to \{a\Gamma_a\}_{a \in \mathcal{A}}}$$

Fixed point rules:

$$\mu\text{-}l\ \frac{\Gamma, e(\mu X e(X)) \to \Delta}{\Gamma, \mu X e(X) \to \Delta} \qquad \mu\text{-}r\ \frac{\Gamma \to \Delta, e(\mu X e(X))}{\Gamma \to \Delta, \mu X e(X)}$$

$$\nu\text{-}l\ \frac{\Gamma, e(\nu X e(X)) \to \Delta}{\Gamma, \nu X e(X) \to \Delta} \qquad \nu\text{-}r\ \frac{\Gamma \to \Delta, e(\nu X e(X))}{\Gamma \to \Delta, \nu X e(X)}$$

NB: these rules do not guarantee extremality of fixed points!

Definition

- **Preproofs** are generated coinductively from the inference rules.
- A preproof is **cyclic/regular** if it has only finitely many distinct sub-preproofs.
- A **proof** is a preproof where each infinite branch has a 'good formula trace'.

'Good formula traces' given by $\exists$-winning plays on RHS, or $\forall$-winning plays on LHS.

Write CRLL for the class of regular $\widehat{\mathsf{RLL}}$ proofs.

Definition

- **Preproofs** are generated coinductively from the inference rules.
- A preproof is **cyclic/regular** if it has only finitely many distinct sub-preproofs.
- A **proof** is a preproof where each infinite branch has a 'good formula trace'.

'Good formula traces' given by $\exists$-winning plays on RHS, or $\forall$-winning plays on LHS.

Write CRLL for the class of regular $\widehat{\mathsf{RLL}}$ proofs.

Example

a^ω has infinitely many as

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{\nu X(aX) \to i_a}\;{\scriptstyle \nu\text{-}l,\nu\text{-}r}\ \bullet}{a\nu X(aX) \to ai_a}\;{\scriptstyle \mathsf{h}_a}}{a\nu X(aX) \to ai_a + bi'_a}\;{\scriptstyle +\text{-}r,\mathsf{wk}\text{-}r}}{a\nu X(aX) \to i'_a}\;{\scriptstyle \mu\text{-}r}}{\nu X(aX) \to i_a}\;{\scriptstyle \nu\text{-}l,\nu\text{-}r}\ \bullet$$

a^ω does not have finitely many as

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{f_a, \nu Y(aY) \to}\ \circ}{af_a, a\nu Y(aY) \to}\;{\scriptstyle \mathsf{h}_a}\quad \text{(easy)}}{af_a + bf_a + \nu Y(bY), a\nu Y(aY) \to}\;{\scriptstyle +\text{-}l}}{f_a, a\nu Y(aY) \to}\;{\scriptstyle \mu\text{-}l}}{f_a, \nu Y(aY) \to}\;{\scriptstyle \nu\text{-}l}\ \circ}{f_a \cap \nu Y(aY) \to}\;{\scriptstyle \cap\text{-}l}$$

Any ω-word over $\{a, b\}$ has finitely many as or infinitely many as

$$
+\text{-}r\,\dfrac{\mu\text{-}r,\nu\text{-}r\,\dfrac{+\text{-}r\,\dfrac{\nu\text{-}l,\nu\text{-}r,\mu\text{-}r\,\dfrac{+\text{-}r\,\dfrac{+_l\,\dfrac{\mathsf{h}_a\,\dfrac{\dfrac{\vdots}{\top \to f_a, i_a}\;\bullet}{a\top \to af_a, ai_a} \qquad \mathsf{h}_a\,\dfrac{\mu\text{-}r,+\text{-}r\,\dfrac{\dfrac{\vdots}{\top \to af_a, bf_a, b^\omega, i'_a}\;\circ}{\top \to f_a, b^\omega, i'_a}}{b\top \to bf_a, bb^\omega, bi'_a}}{a\top + b\top \to af_a, bf_a, bb^\omega, ai_a, bi'_a}}{a\top + b\top \to af_a, bf_a, bb^\omega, ai_a + bi'_a}}{\top \to af_a, bf_a, b^\omega, i'_a}\;\circ}{\top \to af_a + bf_a + b^\omega, i'_a}}{\top \to f_a, i_a}\;\bullet}{\top \to f_a + i_a}
$$

Key

$$b^\omega := \nu Y(bY)$$
$$i'_a := \mu Y(ai_a + bY)$$

Correctness

$-\circ^\omega$: orange trace good

$-\bullet^\omega$: green trace good

$(- \bullet - \circ)^\omega$: green/blue trace good

Outline

All metalogical results are obtained (eventually) by reduction to evaluation game.

All metalogical results are obtained (eventually) by reduction to evaluation game.

Theorem (Soundness)
If $\widehat{\mathsf{RLL}}$ *has a proof of* $e \to f$ *then* $\mathcal{L}(e) \subseteq \mathcal{L}(f)$.

All metalogical results are obtained (eventually) by reduction to evaluation game.

Theorem (Soundness)
If $\widehat{\mathsf{RLL}}$ has a proof of $e \to f$ then $\mathcal{L}(e) \subseteq \mathcal{L}(f)$.

Proof idea.
Let P be a $\widehat{\mathsf{RLL}}$ preproof of $e \to f$, but $w \in \mathcal{L}(e) \setminus \mathcal{L}(f)$.

- Let $\mathfrak{e}/\mathfrak{a}$ be $\exists/\forall$ winning strategies from $(w, e)/(w, f)$, by adequacy.
- $\mathfrak{e}$, $\mathfrak{a}$ together determine an infinite branch $B_{\mathfrak{e},\mathfrak{a}}$ of P:
  - $\mathfrak{e}$ decides the direction at a left branching rule;
  - $\mathfrak{a}$ decides the direction at a right branching rule.
- By construction, any LHS / RHS trace of $B_{\mathfrak{e},\mathfrak{a}}$ is a play of $\mathfrak{e}/\mathfrak{a}$.
  $\therefore$ so $B_{\mathfrak{e},\mathfrak{a}}$ has no good trace. $\square$

All metalogical results are obtained (eventually) by reduction to evaluation game.

Theorem (Soundness)
If $\widehat{\mathsf{RLL}}$ has a proof of $e \to f$ then $\mathcal{L}(e) \subseteq \mathcal{L}(f)$.

Proof idea.
Let P be a $\widehat{\mathsf{RLL}}$ preproof of $e \to f$, but $w \in \mathcal{L}(e) \setminus \mathcal{L}(f)$.

- Let $\mathfrak{e}/\mathfrak{a}$ be $\exists/\forall$ winning strategies from $(w, e)/(w, f)$, by adequacy.
- $\mathfrak{e}$, $\mathfrak{a}$ together determine an infinite branch $B_{\mathfrak{e},\mathfrak{a}}$ of P:
  - $\mathfrak{e}$ decides the direction at a left branching rule;
  - $\mathfrak{a}$ decides the direction at a right branching rule.
- By construction, any LHS / RHS trace of $B_{\mathfrak{e},\mathfrak{a}}$ is a play of $\mathfrak{e}/\mathfrak{a}$.
  $\therefore$ so $B_{\mathfrak{e},\mathfrak{a}}$ has no good trace. $\square$

Corollary
$\mathsf{CRLL} \vdash e \to f \implies \mathcal{L}(e) \subseteq \mathcal{L}(f)$.

For completeness, we need further combinatorial game theoretic machinery.

For completeness, we need further combinatorial game theoretic machinery.

Construe proof search as a 2-player game between `Prover` and `Refuter`.

By ω-regularity of the correctness condition for $\widehat{\mathsf{RLL}}$:

Lemma (cf. Büchi-Landweber)

The proof search game for $\widehat{\mathsf{RLL}}$ is finite memory determined.

For completeness, we need further combinatorial game theoretic machinery.

Construe proof search as a 2-player game between `Prover` and `Refuter`.

By ω-regularity of the correctness condition for $\widehat{\mathsf{RLL}}$:

Lemma (cf. Büchi-Landweber)

The proof search game for $\widehat{\mathsf{RLL}}$ is finite memory determined.

Theorem (Completeness)

$\mathcal{L}(e) \subseteq \mathcal{L}(f) \implies \mathsf{CRLL} \vdash e \to f$, *for* e, f *guarded*.

For completeness, we need further combinatorial game theoretic machinery.

Construe proof search as a 2-player game between `Prover` and `Refuter`.

By ω-regularity of the correctness condition for $\widehat{\mathsf{RLL}}$:

Lemma (cf. Büchi-Landweber)
The proof search game for $\widehat{\mathsf{RLL}}$ is finite memory determined.

Theorem (Completeness)
$\mathcal{L}(e) \subseteq \mathcal{L}(f) \implies \mathsf{CRLL} \vdash e \to f$, *for* e, f *guarded.*

Proof idea.
Assume there is a `Refuter` winning strategy $\mathfrak{R}$ from $e \to f$.

- Play $\mathfrak{R}$ against a canonical `Prover` strategy to obtain a branch B.
- By guardedness, B determines an infinite word $w_B \in \mathcal{A}^\omega$.
- By construction, the LHSs of B determine an $\exists$ strategy winning from (w_B, e).
- Dually, the RHSs of B determine an $\forall$ strategy winning from (w_B, f).

Thus by adequacy we have $w_B \in \mathcal{L}(e) \setminus \mathcal{L}(f)$. $\square$

Summary:

- We proposed an algebraic syntax for APAs and developed a cyclic proof system.
- We developed the basic theory, and proved soundness and completeness.
  NB: the system CRLL also enjoys effective proof search.

Summary:

- We proposed an algebraic syntax for APAs and developed a cyclic proof system.
- We developed the basic theory, and proved soundness and completeness.
  NB: the system CRLL also enjoys effective proof search.

Part of an ongoing programme developing an *algebraic proof theory of formal languages* [DP17, DP18, DDP18, DD24b, DD24a, DD25].

Conclusions

Summary:

- We proposed an algebraic syntax for APAs and developed a cyclic proof system.
- We developed the basic theory, and proved soundness and completeness.
  NB: the system CRLL also enjoys effective proof search.

Part of an ongoing programme developing an *algebraic proof theory of formal languages* [DP17, DP18, DDP18, DD24b, DD24a, DD25].

Other directions:

- In parallel work, we axiomatised the RLL theory of $\mathcal{L}$.
  NB: the principles (1)-(5) are *not* enough! No known q.f. axiomatisation.

Conclusions

Summary:

- We proposed an algebraic syntax for APAs and developed a cyclic proof system.
- We developed the basic theory, and proved soundness and completeness.
  NB: the system CRLL also enjoys effective proof search.

Part of an ongoing programme developing an *algebraic proof theory of formal languages* [DP17, DP18, DDP18, DD24b, DD24a, DD25].

Other directions:

- In parallel work, we axiomatised the RLL theory of $\mathcal{L}$.
  NB: the principles (1)-(5) are *not* enough! No known q.f. axiomatisation.
- What about computational interpretations of CRLL (and friends)?
  NB: beware non-constructivity!

Summary:

- We proposed an algebraic syntax for APAs and developed a cyclic proof system.
- We developed the basic theory, and proved soundness and completeness.
  NB: the system CRLL also enjoys effective proof search.

Part of an ongoing programme developing an *algebraic proof theory of formal languages* [DP17, DP18, DDP18, DD24b, DD24a, DD25].

Other directions:

- In parallel work, we axiomatised the RLL theory of $\mathcal{L}$.
  NB: the principles (1)-(5) are *not* enough! No known q.f. axiomatisation.
- What about computational interpretations of CRLL (and friends)?
  NB: beware non-constructivity!

THANK YOU.

References I

Anupam Das and Abhishek De.
A proof theory of (ω-)context-free languages, via non-wellfounded proofs.
In Christoph Benzmüller, Marijn J. H. Heule, and Renate A. Schmidt, editors, *Automated Reasoning - 12th International Joint Conference, IJCAR 2024, Nancy, France, July 3-6, 2024, Proceedings, Part II*, volume 14740 of *Lecture Notes in Computer Science*, pages 237–256. Springer, 2024.

Anupam Das and Abhishek De.
A proof theory of right-linear (ω-)grammars via cyclic proofs.
In Pawel Sobocinski, Ugo Dal Lago, and Javier Esparza, editors, *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2024, Tallinn, Estonia, July 8-11, 2024*, pages 30:1–30:14. ACM, 2024.

Anupam Das and Abhishek De.
Right-linear lattices: An algebraic theory of ω-regular languages, with fixed points.
In Pawel Gawrychowski, Filip Mazowiecki, and Michal Skrzypczak, editors, *50th International Symposium on Mathematical Foundations of Computer Science, MFCS 2025, August 25-29, 2025, Warsaw, Poland*, volume 345 of *LIPIcs*, pages 39:1–39:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025.

Anupam Das, Amina Doumane, and Damien Pous.

Left-handed completeness for kleene algebra, via cyclic proofs.

In Gilles Barthe, Geoff Sutcliffe, and Margus Veanes, editors, *LPAR-22. 22nd International Conference on Logic for Programming, Artificial Intelligence and Reasoning*, volume 57 of *EPiC Series in Computing*, pages 271–289, Awassa, Ethiopia, 2018. EasyChair.

Christian Dax, Martin Hofmann, and Martin Lange.

A proof system for the linear time μ-calculus.

In S. Arun-Kumar and Naveen Garg, editors, *FSTTCS 2006: Foundations of Software Technology and Theoretical Computer Science, 26th International Conference, Kolkata, India, December 13-15, 2006, Proceedings*, volume 4337 of *Lecture Notes in Computer Science*, pages 273–284. Springer, 2006.

Anupam Das and Damien Pous.

A cut-free cyclic proof system for kleene algebra.

In Renate A. Schmidt and Cláudia Nalon, editors, *Automated Reasoning with Analytic Tableaux and Related Methods - 26th International Conference, TABLEAUX 2017, Brasília, Brazil, September 25-28, 2017, Proceedings*, volume 10501 of *Lecture Notes in Computer Science*, pages 261–277, Brazil, 2017. Springer.

References III

Anupam Das and Damien Pous.

Non-Wellfounded Proof Theory For (Kleene+Action)(Algebras+Lattices).

In Dan R. Ghica and Achim Jung, editors, *27th EACSL Annual Conference on Computer Science Logic (CSL 2018)*, volume 119 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 19:1–19:18, Dagstuhl, Germany, 2018. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

Roope Kaivola.

Axiomatising linear time mu-calculus.

In Insup Lee and Scott A. Smolka, editors, *CONCUR '95: Concurrency Theory, 6th International Conference, Philadelphia, PA, USA, August 21-24, 1995, Proceedings*, volume 962 of *Lecture Notes in Computer Science*, pages 423–437. Springer, 1995.

Damian Niwiński and Igor Walukiewicz.

Games for the μ-calculus.

Theoretical Computer Science, 163(1):99–116, 1996.