For me context is the key - from that comes the understanding of everything. - Kenneth Noland

One-pass Context-based Tableaux Systems for CTL and ECTL.
TIME 2020

Alex Abuin
Alexander Bolotov
Montserrat Hermo
Paqui Lucio

25 September 2020
Structure of the Talk

- Motivation/General Problem Structure
- Branching-Time Logics Framework
- Tableaux: General Idea
- More motivation: 1 pass vs 2 pass Tableaux
- Our Approach - Context Based Tableaux
- Formalisms/Tableaux Rules for CTL
- Extension to ECTL
- Conclusions and Future Work
Motivation

General Problem Structure

System \rightarrow System Specification \rightarrow Verification \rightarrow Deduction

- Automata Emptiness
- Resolution
- Tableaux
- Sequents
Motivation

General Problem Structure

- System
- System Specification
- Verification
- Deduction
  - Automata Emptiness
  - Resolution
  - Tableaux
  - Sequents
    - Two-Pass Tableaux
    - One-Pass Tableaux

Duality
Recovered
Branching-Time Framework

- Temporal operators:
  - ◦ - ‘at the next moment of time’, ◊ - ‘eventually’, □ - ‘always’
  and $\mathcal{U}$ - ‘until’
- paths quantifiers: A - ‘for all paths’; E - ‘there exists a path

$S_0 \models A \Box b$

$S_0 \models E \Diamond c$

$S_0 \models E(\neg c \mathcal{U} d)$
Tableaux: very general idea for one and two pass methods

- Labelled graph
- Expansion rules: $\alpha$ rules generate a successor node $\frac{\phi \land \psi}{\phi, \psi}$
- $\beta$ rules ‘split’ into alternative nodes $\frac{\phi \lor \psi}{\phi \mid \psi}$
- Use fixpoint properties of temporal operators:
  \[ A\Box \phi = \phi \land A\Diamond A\Box \phi \quad E\phi U\psi = \psi \lor (\phi \land E\Diamond E\phi U\psi) \]

\[
\begin{align*}
A\Box \phi & \quad \frac{A\Box \phi}{\phi, A\Diamond A\Box \phi} \\
E\phi U\psi & \quad \frac{E\phi U\psi}{\psi \mid \phi \land E\Diamond E\phi U\psi}
\end{align*}
\]
A state is a node such that all its formulae are literals or of the form $A \phi$ or $E \phi$

This allows to build successor nodes to propagate $\phi$.

In our construction the tableaux graph is an AND-OR TREE:
Motivation: 2 pass vs 1 pass

- **Two-pass tableaux:**
  1st pass: checks the validity of the given formula by creating a graph
  2nd pass: checks if all eventualities are satisfied.

- **One-pass tableaux:**
  one pass: checks the validity of the input without auxiliary constructions,
  need: mechanism to ensure this.

- Our mechanism to ensure a one pass - CONTEXT (of the selected eventuality)
Branching-Time Logics Framework: context

- CTL well formed formulae are state formulae - every temporal operator is preceded by a path quantifier:
  - E◊p ✓
  - E□◊p X

- Context
  - is a collection of state formulae accompanying the eventuality in the node.
  - forces eventualities to be fulfilled as soon as possible.

- In the node \{ E(pUq), p \} the context of E(pUq) is p
CONTEXT forces fulfilment of eventualities

\[ E(p \cup q), p \]

Expand using the context based rule

\[ p, \text{ EoE } [(p \land \neg p) \cup q] \]

Construct a pre-state

\[ E[(p \land \neg p) \cup q] \]

\[ E[A \cup B] = B \lor A \land \text{EoE}[A \cup B] \]

q

Model: \{p,q\}

\[ p \land \neg p, \text{ EoE } [(p \land \neg p) \cup q] \]

No. model: contradiction

Model: \{q\}
Branching-Time Logics Framework: CTL-type logics

State formulae

\[ \sigma ::= t \mid p \mid \sigma_1 \land \sigma_2 \mid \neg \sigma \mid E\pi \]

Path formulae

\[ \begin{align*}
\pi^{\text{CTL}*} & ::= \sigma \mid \pi_1 \land \pi_2 \mid \neg \pi \mid \diamond \pi \mid \pi U \pi \\
\pi^{\text{ECTL}#} & ::= \sigma \mid \pi_1 \land \pi_2 \mid \neg \pi \mid \diamond \pi \mid \sigma U (\sigma \land \lozenge \sigma) \\
& \quad \mid \square (\pi \lor \square \pi) \mid \sigma U (\square \pi) \mid \square (\sigma U \pi) \\
\pi^{\text{ECTL}^+} & ::= \sigma \mid \pi_1 \land \pi_2 \mid \neg \pi \mid \diamond \pi \mid \sigma U \pi \mid \square \pi \mid \square \diamond \pi \mid \diamond \square \pi. \\
\pi^{\text{ECTL}} & ::= \sigma \mid \neg \pi \mid \diamond \pi \mid \sigma U \pi \mid \square \pi \mid \square \diamond \pi \mid \diamond \square \pi. \\
\pi^{\text{CTL}} & ::= \sigma \mid \neg \pi \mid \diamond \pi \mid \sigma U \pi \mid \square \pi. 
\end{align*} \]
Development of One Pass Method for Branching Time

- Previous work - one pass tableaux for ECTL
- This covers ECTL
- However, too complex to cover simpler logics - CTL and ECTL
- We made a ‘U-turn’ from ECTL (and ECTL+) to bridge this gap

<table>
<thead>
<tr>
<th>BTL Logics</th>
<th>$E\Box q$</th>
<th>$E(\Box q \land \Box \neg q)$</th>
<th>$A((p\cup \Box q) \lor (s\cup \Box \neg q))$</th>
<th>$A(\Box (\neg p \land E \neg p))$</th>
<th>One-pass Tableaux</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B(U, o)$ (CTL)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>This paper</td>
</tr>
<tr>
<td>$B(U, o, \Box)$ (ECTL)</td>
<td>\checkmark</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>This paper</td>
</tr>
<tr>
<td>$B^+(U, o, \Box)$ (ECTL+)</td>
<td>\checkmark</td>
<td>\checkmark</td>
<td>X</td>
<td>X</td>
<td>✓</td>
</tr>
<tr>
<td>$B^+(U, o, U\Box)$ (ECTL#)</td>
<td>\checkmark</td>
<td>\checkmark</td>
<td>✓</td>
<td>X</td>
<td>✓</td>
</tr>
<tr>
<td>$B^+(U, o)$ (CTL*)</td>
<td>\checkmark</td>
<td>\checkmark</td>
<td>✓</td>
<td>✓</td>
<td>X</td>
</tr>
</tbody>
</table>
One-pass Tableau Method for CTL - \( \alpha \)- and \( \beta \)-Rules

- Standard but preserve context

\[
\begin{align*}
\land & : \quad \frac{\Sigma, \sigma_1 \land \sigma_2}{\Sigma, \sigma_1, \sigma_2} \\
\lor & : \quad \frac{\Sigma, \sigma_1 \lor \sigma_2}{\Sigma, \sigma_1 \mid \Sigma, \sigma_2} \\
\text{(Q } & \mathcal{R} \text{)} : \quad \frac{\Sigma, \text{Q}(\sigma_1 \mathcal{R} \sigma_2)}{\Sigma, \sigma_1, \sigma_2 \mid \Sigma, \sigma_2, \text{Q} \circ \text{Q}(\sigma_1 \mathcal{R} \sigma_2)} \\
\text{(Q} & \Box \text{) : } \quad \frac{\Sigma, \text{Q} \Box \sigma}{\Sigma, \sigma, \text{Q} \circ \text{Q} \Box \sigma} \\
\text{(Q } & \mathcal{U} \text{) : } \quad \frac{\Sigma, \text{Q}(\sigma_1 \mathcal{U} \sigma_2)}{\Sigma, \sigma_2 \mid \Sigma, \sigma_1, \text{Q} \circ \text{Q}(\sigma_1 \mathcal{U} \sigma_2)} \\
\text{(Q} & \diamond \text{) : } \quad \frac{\Sigma, \text{Q} \diamond \sigma}{\Sigma, \sigma \mid \Sigma, \text{Q} \circ \text{Q} \diamond \sigma}
\end{align*}
\]
One-pass Tableau Method for CTL: Next-state Rule

- Responsible for obtaining AND-OR tree

\[
(Q_\circ) \quad \frac{\Sigma, A\circ\sigma_1, \ldots, A\circ\sigma_\ell, E\circ\sigma'_1, \ldots, E\circ\sigma'_k, \sigma_1, \ldots, \sigma_\ell, \sigma'_1 \& \ldots \& \sigma_1, \ldots, \sigma_\ell, \sigma'_k}{\sigma_1, \ldots, \sigma_\ell, \sigma'_1 \& \ldots \& \sigma_1, \ldots, \sigma_\ell, \sigma'_k}
\]

where \( \Sigma \) is a set of literals.

\[
A\circ A(F R \neg q), E\circ E(p U q) \land E\circ \neg q
\]

\[
A\circ A(F R \neg q), E\circ E(p U q), E\circ \neg q
\]

\[
A(F R \neg q), E(p U q) \quad A(F R \neg q), \neg q
\]
One-pass Tableau Method for CTL: CONTEXT

- \( \beta^+ \)-Rule – manages CONTEXT

\[
\frac{(QU)^+}{\Sigma, Q(\sigma_1 U \sigma_2)} \quad \frac{\Sigma, \sigma_2}{\Sigma, \sigma_1, Q \circ Q((\sigma_1 \land \sim \Sigma') U \sigma_2)}
\]
where
\[
\Sigma' = \Sigma \setminus \{ (A \circ)^i A \boxdot \sigma \mid i \geq 0 \text{ and } (A \circ)^i A \boxdot \sigma \in \Sigma \} \text{ and }
\]
\((A \circ)^i\) stands for \(i\) times \(A \circ\).

\[
\begin{align*}
A(F R \neg q), E(p U q) & \quad (EU)^+ \quad A(F R \neg q), q \\
(ERA) & \quad A(F R \neg q), p, E \circ E((p \land E(\tau U q)) U q)
\end{align*}
\]
**Tableaux Algorithm** $\tau(\sigma)$

1: procedure SYSTEMATIC_TABLEAU($\Sigma_0$)  
2: \quad if $\Sigma_0$ is not uniform then $T :=$ Uniform_Tableau($\Sigma_0$)  
3: \quad while $T$ has at least one expandable node do  
4: \quad \quad \triangleright$Invariant$: Any expandable node of $T$ is labelled by an uniform set  
5: \quad \quad Choose any node $\ell$ in $T$ such that $\tau(\ell)$ is expandable  
6: \quad \quad \triangleright$\tau(\ell)$ is uniform  
7: \quad \quad if there is no eventuality in $\tau(\ell)$ then $T := T[\ell \leftarrow \text{Uniform}_\text{Tableau}(\tau(\ell))]$  
8: \quad \quad else  
9: \quad \quad \quad Eventuality\_Selection(\tau(\ell))  
10: \quad \quad \quad Apply\_\beta^+-\text{rule}(\tau(\ell))  
11: \quad \quad \quad Let $\ell_1, \ell_2$ the two children of $\ell$  
12: \quad \quad \quad for $i = 1 \ldots 2$ do  
13: \quad \quad \quad \quad if $\ell_i$ is expandable and $\tau(\ell_i)$ is not uniform then  
14: \quad \quad \quad \quad \quad $T := T[\ell_i \leftarrow \text{Uniform}_\text{Tableau}(\tau(\ell_i))]$
Tableau Extension to ECTL

Two additional rules

\[
\begin{align*}
(Q \Box \Diamond) & \quad \frac{\Sigma, Q \Box \Diamond \sigma}{\Sigma, Q \Diamond \sigma, Q \circ Q \Box \Diamond \sigma} \\
(Q \Diamond \Diamond) & \quad \frac{\Sigma, Q \Diamond \Diamond \sigma}{\Sigma, Q \Diamond \sigma, Q \circ Q \Box \Diamond \sigma}
\end{align*}
\]

that are based on these logical equivalences:

\[
\begin{align*}
E \Box \Diamond \sigma & \equiv E \Diamond \sigma \land E \circ E \Box \Diamond \sigma \\
A \Box \Diamond \sigma & \equiv A \Diamond \sigma \land A \circ A \Box \Diamond \sigma \\
E \Box \Diamond \sigma & \equiv E \Diamond \sigma \lor (E \Diamond \sigma \land E \circ E \Box \Diamond \sigma) \\
A \Box \Diamond \sigma & \equiv A \Diamond \sigma \lor (A \Diamond \sigma \land A \circ A \Box \Diamond \sigma)
\end{align*}
\]
Conclusions and Future Work

- Context-based tableau extends naturally from linear-time to branching-time.
  - CTL tableau is much simpler than ECTL, one, where two types of context (outer and inner) are used.
- Context-based tableau keeps the duality with sequent calculus.
  - Application in certified model checking: sequent proofs as certificates.
- Intuitive tableau method for CTL and ECTL.
  - Well suited for the automation and amenable for the implementation.