Soundness of LTL Model Checking

Aditya Kanade

E0223, CSA, IISc.
## LTL Model Checking

**TS |= \varphi?**

<table>
<thead>
<tr>
<th>Step</th>
<th>Algorithm</th>
<th>Soundness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>LTL $\rightarrow$ GNBA</td>
<td>$Lw(G_{\varphi}) = \text{words}(\varphi)$</td>
</tr>
<tr>
<td>2.</td>
<td>GNBA $\rightarrow$ NBA</td>
<td>$Lw(G_{\varphi}) = Lw(A_{\varphi})$</td>
</tr>
<tr>
<td>3.</td>
<td>TS $\otimes A$</td>
<td>Product construction</td>
</tr>
<tr>
<td>4.</td>
<td>TS $\otimes A \models \square \square T F$ (persistence checking)</td>
<td>Nested DFS</td>
</tr>
</tbody>
</table>
LTL to GNBA

To prove that $L\omega(G\varphi) = \text{words}(\varphi)$.

2: Let $\bar{s} = A_0 A_1 A_2 \ldots \in \text{words}(\varphi)$.

(a) There exists a run $\overline{s} = B_0 B_1 B_2 \ldots$ of $G\varphi$ on $\bar{s}$.

(b) The run $\overline{s}$ is an accepting run of $G\varphi$.

$\subseteq$: If $\bar{s} = A_0 A_1 A_2 \ldots \in L\omega(G\varphi)$ then $\bar{s} \models \varphi$. 
\[ L_w(g, \gamma) \geq \text{words}(\gamma) \]

Let \( \gamma = A_0 A_1 A_2 \ldots \in \text{words}(\gamma) \).

(a) Let \( B_i = \{ \psi \in \text{closure}(\gamma) \mid A_i A_{i+1} \ldots = \psi \} \).

- \( B_i \) is an elementary set of formulae i.e. \( B_i \in \wp \).
- \( B_{i+1} \subseteq f(B_i, A_i) \), for all \( i \geq 0 \).
  - \( A_i = B_i \cap \text{AP} \Rightarrow f(B_i, A_i) \neq \emptyset \)
  - \( \psi \in B_i \) iff \( \psi \in B_{i+1} \) \( \sim \) Def. of \( f \) in the construction
  - \( \psi_1 \cup \psi_2 \in B_i \) iff \( (\psi_2 \in B_i) \lor (\psi_1 \in B_i \text{ and } \psi_1 \cup \psi_2 \in B_{i+1}) \) 
    \( \sim \) Def. of \( f \) in the construction
(b) Prove that \( B_i \in F(\psi_1 \cup \psi_2) \), for infinitely many \( i \), for all \( \psi_1 \cup \psi_2 \in \text{closure}(\psi) \).

Let there be finitely many \( j \) such that \( B_j \in F(\psi_1 \cup \psi_2) \).

\[ B_i \notin F(\psi_1 \cup \psi_2) \Rightarrow \psi_1 \cup \psi_2 \notin B_i \text{ and } \psi_2 \notin B_i \ldots \text{(construction)} \]

Now, \( A_i A_{i+1} \ldots \models \psi_1 \cup \psi_2 \text{ and } A_i A_{i+1} \ldots \not\models \psi_2 \ldots \text{(Def. of } B_i) \)

Hence, there exists \( k \geq i \), \( A_k A_{k+1} \ldots \models \psi_2 \).

Thus, \( \psi_2 \in B_k \). Further, \( B_k \in F(\psi_1 \cup \psi_2) \).

If \( B_i \notin F(\psi_1 \cup \psi_2) \) for \( i \neq m \), then \( B_k \in F(\psi_1 \cup \psi_2) \) for \( i = m \). \( k \).

Hence, \( G = B_0 B_1 B_2 \ldots \) is an accepting run of \( G_\psi \).
Let $6 = A_0 A_1 A_2 \ldots \in \text{Lw}(G \psi)$.

Let $B_0 B_1 B_2 \ldots$ be the corresponding accepting run of $G \psi$.

We have $A_i = B_i \cap \text{AP}$ and

$6 = (B_0 \cap \text{AP}) (B_1 \cap \text{AP}) (B_2 \cap \text{AP}) \ldots = \psi$?

Prove that,

for all $\psi \in \text{closure}(\psi)$,

$\psi \in B_0 \iff A_0 A_1 A_2 \ldots = \psi$. 
Proof by structural induction on the structure of $\psi$.

**Base case:** $\psi = \text{true}^*$, $\psi \in A^P$ (*Ref. to additional slides*)

**Induction step:** $\psi_1 \land \psi_2^*$, $\Diamond \psi$, $\psi_1 \lor \psi_2$

1. If $\psi_1 \lor \psi_2 \in B_0$ then $\Diamond \bigoplus A_1 A_2 \ldots = \psi_1 \lor \psi_2$.
   
   $\psi_1 \in B_0$ or $\psi_2 \in B_0$.

   *Let $\psi_2 \notin B_j$, for all $j \geq 0$.

   $\psi_1 \in B_j$ and $\psi_1 \lor \psi_2 \in B_j$, for $j \geq 0$ (by construction).

   However, $B_0 B_1 B_2 \ldots$ is accepting.
Therefore, $B_j \in F$ for i.m. $j \geq 0$.

\[(y_1 \cup y_2)\]

But, we just showed that,

\[y_2 \not\in B_j \text{ and } y_1 \cup y_2 \in B_j\]

iff

\[B_j \not\in F(y_1 \cup y_2), \text{ for all } j ... (by \ construction)\]

Contradiction.

Hence, $y_2 \in B_j$ and $y_1 \in B_i$, $0 \leq i < j$ ... (by construction)

By hypothesis, $A_j \ldots = y_2$, $A_i \ldots = y_1$, $0 \leq i < j$.

Hence, $A_0 A_1 \ldots = y_1 \cup y_2$. 
2. If \( A_0 A_1 \ldots \subseteq \psi_1 \cup \psi_2 \) then \( \psi_1 \cup \psi_2 \subseteq B_0 \).

Let \( A_0 A_1 \cdots \subseteq \psi_1 \cup \psi_2 \). There exists a \( j \) s.t.

\[ \psi_2 \subseteq B_j \quad \text{(by induction hypothesis)} \]

\[ \psi_1 \subseteq B_i, \quad 0 \leq i < j \]

By the definition of elementary sets:

\[ \psi_1 \cup \psi_2 \subseteq B_j \]

\[ \psi_1 \cup \psi_2 \subseteq B_i, \quad 0 \leq i < j \]

Hence, proved.
To prove that $Lw(\text{G}_p) = Lw(\text{A}_p)$

Let $F$ be the acc. set of $\text{A}_p$ and $F_i, \ldots, F_k$ be acc. sets of $\text{G}_p$.

To prove:

2: Let $w \in Lw(\text{A}_p)$ and $\gamma$ be the corr. acc. run.

$$\text{Inf}(\gamma) \cap F \neq \emptyset \iff \text{Inf}(\gamma) \cap F_i \neq \emptyset, \text{ for all } i.$$ 

Thus, $w \in Lw(\text{G}_p)$.

\[ \leq \]

3: Let $w \in Lw(\text{G}_p)$. Suppose $w \notin Lw(\text{A}_p)$.

Let the run $\gamma$ of $A$ on $w$ be stuck in an $i'$th copy.

Thus, $\text{Inf}(\gamma) \cap F_i = \emptyset$. Otherwise, you escape.

The run $\gamma$ corr. to a run $\gamma'$ of $\text{G}_p$ (on the first comp.). Contradiction.
Soundness of Nested DFS

To prove that the nested DFS does not miss a cycle containing S ≠ a (even though we ignore the states visited in previous calls to CYCLE-CHECK.)

To prove that upon calling CYCLE-CHECK(s), there is no cycle of the form s t₁ ... tₖ s such that some tᵢ ∈ Y, 1 ≤ i ≤ k.

↑
global set of visited states
Suppose there is some $t_i = t$ s.t. $t \in V$. There must be some $u \neq a$ s.t. $t$ was visited during a call $\text{CYCLECHECK}(u) < \text{CYCLECHECK}(s)$. We therefore have the following reachability relations:

```
\[ \sim \quad u \neq a \quad \circ \quad t = a \quad \circ \quad s \neq a \quad \text{visited} \]
```
(a) Let \( u \) precede \( s \) in the outer DFS.

Hence, \( \text{CYCLE-CHECK}(s) < \text{CYCLE-CHECK}(u) \).

Contradiction with *

(b) Let \( s \) precede \( u \) in the outer DFS.

Hence, \( u \) is reachable from \( s \) ...(as \( s \) was on the stack.)

Already, \( t \) is reachable from \( u \) and

\[ s \text{ is reachable from } t \]

\[ \implies s \text{ is reachable from } u. \]

This (or some other) cycle containing \( u \) should be
detected & no call of \( \text{CYCLE-CHECK}(s) \).