CS 228 : Logic in Computer Science

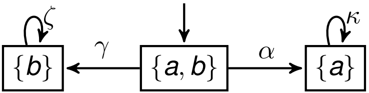
Krishna. S

LTL ModelChecking

- Given transition system *TS*, and LTL formula φ, does $TS \models \varphi$?
- $Tr(TS) \subseteq L(\varphi)$ iff $Tr(TS) \cap \overline{L(\varphi)} = \emptyset$
- First construct NBA $A_{\neg\varphi}$ for $\neg\varphi$.
- Construct product of TS and $A_{\neg\varphi}$, obtaining a new TS, say TS'.
- Check some very simple property on TS', to check $TS \models \varphi$.

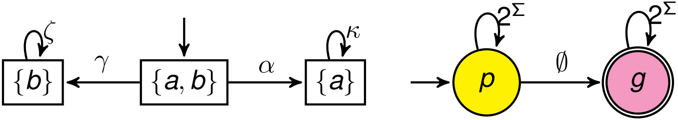
An Example $TS \models \varphi$

- Let $\varphi = \square(a \vee b), \neg\varphi = \Diamond(\neg a \wedge \neg b)$
- Take TS and $A_{\neg\varphi}$, and check the intersection.



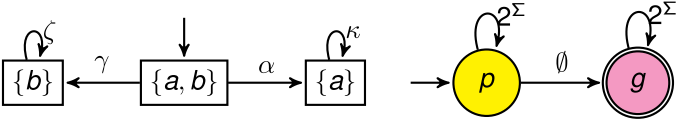
An Example $TS \models \varphi$

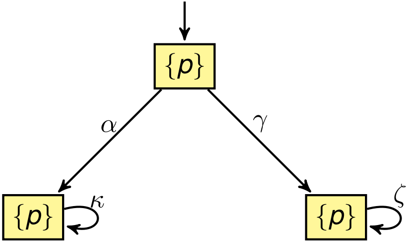
- Let $\varphi = \Box(a \vee b)$, $\neg\varphi = \Diamond(\neg a \wedge \neg b)$
- Take TS and $A_{\neg\varphi}$, and check the intersection.



An Example $TS \models \varphi$

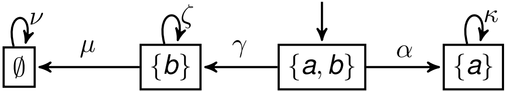
- Let $\varphi = \Box(a \vee b)$, $\neg\varphi = \Diamond(\neg a \wedge \neg b)$
- Take TS and $A_{\neg\varphi}$, and check the intersection.





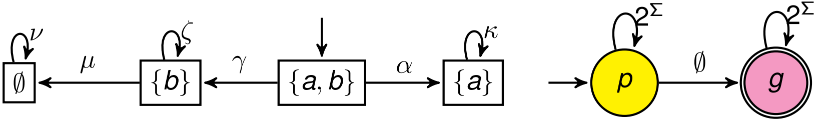
An Example : $TS \not\models \varphi$

- Let $\varphi = \square(a \vee b), \neg\varphi = \Diamond(\neg a \wedge \neg b)$
- Take TS and $A_{\neg\varphi}$, and check the intersection.



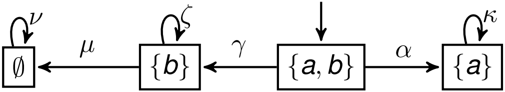
An Example : $TS \not\models \varphi$

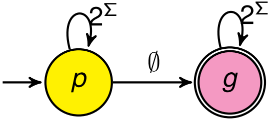
- Let $\varphi = \Box(a \vee b)$, $\neg\varphi = \Diamond(\neg a \wedge \neg b)$
- Take TS and $A_{\neg\varphi}$, and check the intersection.

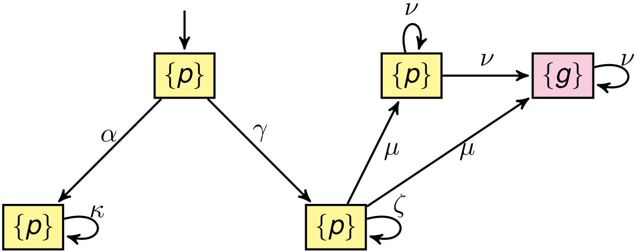


An Example : $TS \not\models \varphi$

- Let $\varphi = \Box(a \vee b), \neg\varphi = \Diamond(\neg a \wedge \neg b)$
- Take TS and $A_{\neg\varphi}$, and check the intersection.







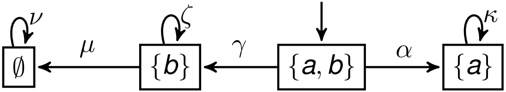
Product of TS and NBA

Given $TS = (S, Act, I, AP, L)$ and $\mathcal{A} = (Q, 2^{AP}, \delta, Q_0, G)$ NBA.
Define $TS \otimes \mathcal{A} = (S \times Q, Act, I', AP', L')$ such that

- $I' = \{(s_0, q) \mid s_0 \in I \text{ and } \exists q_0 \in Q_0, q_0 \stackrel{L(s_0)}{\rightarrow} q\}$
- $AP' = Q$, $L' : S \times Q \rightarrow 2^Q$ such that $L'((s, q)) = \{q\}$
- If $s \stackrel{\alpha}{\rightarrow} t$ and $q \stackrel{L(t)}{\rightarrow} p$, then $(s, q) \stackrel{\alpha}{\rightarrow} (t, p)$

Persistence Properties

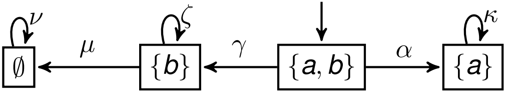
Let η be a propositional logic formula over AP. A persistence property P_{pers} has the form $\Diamond\Box\eta$. How will you check a persistence property on a TS?



- For example, $TS \not\models \Diamond\Box(a \vee b)$
- For example, $TS \models \Diamond\Box(a \vee (a \rightarrow b))$

Persistence Properties

Let η be a propositional logic formula over *AP*. A persistence property P_{pers} has the form $\Diamond\Box\eta$. How will you check a persistence property on a TS?



- For example, $TS \not\models \Diamond\Box(a \vee b)$
- For example, $TS \models \Diamond\Box(a \vee (a \rightarrow b))$
- $TS \not\models P_{pers}$ iff there is a reachable cycle in the TS containing a state with a label which satisfies $\neg\eta$.

LTL ModelChecking

- Given TS and LTL formula φ. Does $TS \models \varphi$?
- Construct $A_{\neg\varphi}$, and let $g_1, \ldots, g_n$ be the good states in $A_{\neg\varphi}$.
- Build $TS' = TS \otimes A_{\neg\varphi}$.
- The labels of TS' are the state names of $A_{\neg\varphi}$.
- Check if $TS' \models \Diamond\Box(\neg g_1 \wedge \ldots \neg g_n)$.

LTL ModelChecking

- Given TS and LTL formula φ. Does $TS \models \varphi$?
- Construct $A_{\neg\varphi}$, and let $g_1, \ldots, g_n$ be the good states in $A_{\neg\varphi}$.
- Build $TS' = TS \otimes A_{\neg\varphi}$.
- The labels of TS' are the state names of $A_{\neg\varphi}$.
- Check if $TS' \models \Diamond\Box(\neg g_1 \wedge \ldots \neg g_n)$.

ModelChecking LTL in TS = Check Persistence in TS'

The following are equivalent.

- $TS \models \varphi$
- $Tr(TS) \cap L(A_{\neg\varphi}) = \emptyset$
- $TS' \models \Diamond\Box(\neg g_1 \wedge \ldots \neg g_n)$.

Complexity of LTL Modelchecking

- Given φ, $A_{\neg\varphi}$ has $\leqslant 2^{|\varphi|}$ states
- $TS \otimes A_{\neg\varphi}$ has $\leqslant |TS|.2^{|\varphi|}$ states
- Persistence checking : Checking $\Box\Diamond\eta$ on $TS \otimes A_{\neg\varphi}$ takes time linear in $\eta.|TS \otimes A_{\neg\varphi}|$

A Weak Lower Bound

The hamiltonian path problem is polynomially reducible to the complement of the LTL modelchecking problem.

- Given graph $G = (V, E)$ synthesize in polynomial time a TS and an LTL formula φ
- Show that G has a HP iff $TS \not\models \varphi$

A Weak Lower Bound

The hamiltonian path problem is polynomially reducible to the complement of the LTL modelchecking problem.

- Given graph $G = (V, E)$ synthesize in polynomial time a TS and an LTL formula φ
- Show that G has a HP iff $TS \not\models \varphi$
- TS is the graph itself, with one new node added, say b such all vertices of G have an edge to b, and b has a self loop. Let the label of a node in the TS be the name of the vertex.
- Write an LTL formula to capture absence of a HP in G. Assume $V = \{v_1, \ldots, v_n\}$.
- The formula $\varphi = \neg\psi$ where ψ is

$$(\Diamond v_1 \wedge \Box(v_1 \rightarrow \bigcirc\Box\neg v_1)) \wedge \ldots (\Diamond v_n \wedge \Box(v_n \rightarrow \bigcirc\Box\neg v_n))$$

- Show that G has a HP iff $TS \not\models \varphi$.

A Weak Lower Bound

Assume *TS* $\not\models \neg\psi$. Then there is a path witnessing ψ.

- Let π be the path in *TS* such that $\pi \models \psi$.

A Weak Lower Bound

Assume $TS \not\models \neg\psi$. Then there is a path witnessing ψ.

- Let π be the path in TS such that $\pi \models \psi$.
- As $\pi \models \bigwedge_{v \in V}(\Diamond v \wedge \Box(v \rightarrow \bigcirc\Box\neg v))$, π witnesses all vertices of V, and does not repeat any vertex.

A Weak Lower Bound

Assume $TS \not\models \neg\psi$. Then there is a path witnessing ψ.

- Let π be the path in TS such that $\pi \models \psi$.
- As $\pi \models \bigwedge_{v \in V}(\Diamond v \wedge \Box(v \rightarrow \bigcirc\Box\neg v))$, π witnesses all vertices of V, and does not repeat any vertex.
- π has the form $v_{i_1} v_{i_2} \ldots v_{i_n} b^\omega$, $i_1, \ldots, i_n \in \{1, 2, \ldots, n\}$, $i_j \neq i_k$.

A Weak Lower Bound

Assume $TS \not\models \neg\psi$. Then there is a path witnessing ψ.

- Let π be the path in TS such that $\pi \models \psi$.
- As $\pi \models \bigwedge_{v \in V}(\Diamond v \wedge \Box(v \rightarrow \bigcirc\Box\neg v))$, π witnesses all vertices of V, and does not repeat any vertex.
- π has the form $v_{i_1} v_{i_2} \ldots v_{i_n} b^\omega$, $i_1, \ldots, i_n \in \{1, 2, \ldots, n\}$, $i_j \neq i_k$.
- So G has the HP $v_{i_1} v_{i_2} \ldots v_{i_n}$.

A Weak Lower Bound

Assume $TS \not\models \neg\psi$. Then there is a path witnessing ψ.

- Let π be the path in TS such that $\pi \models \psi$.
- As $\pi \models \bigwedge_{v \in V}(\Diamond v \wedge \Box(v \rightarrow \bigcirc\Box\neg v))$, π witnesses all vertices of V, and does not repeat any vertex.
- π has the form $v_{i_1} v_{i_2} \ldots v_{i_n} b^\omega$, $i_1, \ldots, i_n \in \{1, 2, \ldots, n\}$, $i_j \neq i_k$.
- So G has the HP $v_{i_1} v_{i_2} \ldots v_{i_n}$.
- The converse is similar : a HP in G extends to a path $\pi = v_{i_1} v_{i_2} \ldots v_{i_n} b^\omega$ in TS. Clearly, $\pi \models \psi$.

A Weak Lower Bound

Assume $TS \not\models \neg\psi$. Then there is a path witnessing ψ.

- Let π be the path in TS such that $\pi \models \psi$.
- As $\pi \models \bigwedge_{v \in V}(\Diamond v \wedge \Box(v \rightarrow \bigcirc\Box\neg v))$, π witnesses all vertices of V, and does not repeat any vertex.
- π has the form $v_{i_1} v_{i_2} \dots v_{i_n} b^\omega$, $i_1, \dots, i_n \in \{1, 2, \dots, n\}$, $i_j \neq i_k$.
- So G has the HP $v_{i_1} v_{i_2} \dots v_{i_n}$.
- The converse is similar : a HP in G extends to a path $\pi = v_{i_1} v_{i_2} \dots v_{i_n} b^\omega$ in TS. Clearly, $\pi \models \psi$.
- So LTL model checking is co-NP hard as HP is NP-complete.
- Actual complexity of LTL model checking : PSPACE-complete. For this, show that given a LBTM M and a word w, construct in poly time a TS and an LTL formula φ such that M accepts w iff $TS \models \varphi$.

LTL Summary

- LTL : temporal logic for specification of programs/systems, useful in checking program/system correctness
- Studied modelchecking
- Widely used in industry : SPIN tool for LTL modelchecking

CS 228 : Taking Stock

- Propositional Logic : Formal proofs, soundness, completeness
- FO and MSO : Expressiveness, satisfiability
- LTL : model checking
- Advanced topics for the interested : Book of Baier-Katoen