Overview

1. Some Complexity Classes
2. Complexity of LTL Model Checking
3. Complexity of LTL Satisfiability
4. Summary

LTL Decision Problems

The LTL Model-Checking Problem
Given a finite transition system $TS$ and LTL-formula $\varphi$, is $TS \vDash \varphi$?

The LTL Satisfiability Problem
Given LTL-formula $\varphi$, does there exist a transition system $TS$ such that $TS \vDash \varphi$?

The LTL Validity Problem
Given LTL-formula $\varphi$, does $TS \vDash \varphi$ for all transition systems $TS$?

The validity problem for $\varphi$ is equivalent to the satisfiability problem for $\neg \varphi$
Overview

1. Some Complexity Classes
2. Complexity of LTL Model Checking
3. Complexity of LTL Satisfiability
4. Summary

Complexity Classes P and NP

- Complexity Class co-NP
- Co-NP Completeness

**NP-hard problems**

**P**

**NP**

**NPC**

**coNPC**

\[ NP \text{-hard problems} \]

\[ coNP = \{ \overline{L} : L \in NP \} \]

\[ \text{complement of } L \]

\[ NPC = \text{class of } NP\text{-complete problems} \]

\[ \begin{array}{l}
(1) \quad L \in NP \\
(2) \quad L \text{ is } NP\text{-hard, i.e., } K \leq_{poly} L \text{ for all } K \in NP
\end{array} \]

\[ coNPC = \text{class of coNP-complete problems} \]

\[ \begin{array}{l}
(1) \quad L \in coNP \\
(2) \quad L \text{ is coNP-hard, i.e., } K \leq_{poly} L \text{ for all } K \in coNP
\end{array} \]
Co-NP Hardness

The LTL model-checking problem is co-NP-hard.

Proof.

We show that the Hamiltonian path problem—which is NP-complete—for directed graphs is polynomially reducible to the complement of the LTL model-checking problem. The complement LTL model-checking problem is: given finite transition system $TS$ and LTL-formula $\phi$, is $TS \not\models \phi$?

As the complement LTL model-checking problem is NP-complete, the result follows.

The Complexity Class PSPACE

PSPACE is the set of decision problems that can be solved by a deterministic, polynomially-bounded space algorithm.

Known facts:

- $NP \subseteq PSPACE$
- $PSPACE = \text{co-PSPACE}$ (this holds for any deterministic complexity class)
- $PSPACE = \text{NPSPACE}$ (Savitch theorem)

Due to Savitch’s theorem, to show $L$ is in PSPACE, it suffices to provide a nondeterministic, polynomially-bounded space algorithm for $\overline{L}$. 

PSPACE is the set of decision problems that can be solved by a deterministic, polynomially-bounded space algorithm.
Complexity Landscape

Some Complexity Classes

PSPACE = coPSPACE = NPSPACE:

\( L \) is PSPACE-complete iff

1. \( L \in \text{PSPACE} \)
2. \( L \) is PSPACE-hard \( \iff \) \( \overline{L} \) is PSPACE-hard

The complement LTL model-checking problem is: given finite transition system \( TS \) and LTL-formula \( \phi \), is \( TS \not\models \phi \)? That is, is there some path \( \pi \) in \( TS \) such that \( \pi \not\models \phi \), i.e., \( \pi \models \neg \phi \)?

This is the existential LTL-model-checking problem for \( \psi = \neg \phi \).

Complexity of LTL Model Checking

LTL Model Checking is PSPACE Complete

The LTL model-checking problem is PSPACE-complete.

Proof.

We prove that the existential LTL model-checking problem is (a) in NPSPACE (and thus in PSPACE), and (b) PSPACE-hard.

The complement LTL model-checking problem is: given finite transition system \( TS \) and LTL-formula \( \phi \), is \( TS \not\models \phi \)? That is, is there some path \( \pi \) in \( TS \) such that \( \pi \not\models \phi \), i.e., \( \pi \models \neg \phi \)?

Given:

- finite transition system \( TS \)
- LTL formula \( \phi \)

Q: is there some path \( \pi \) in \( TS \) such that \( \pi \not\models \phi \)?

Goal:

find a criterion for the existence of a path \( \pi \) in \( TS \) with \( \pi \not\models \phi \) that can be checked non-deterministically in poly-space

Idea:

Use the GNBA \( \exists \phi \) for \( \phi \)
Proof Idea

1. Guess nondeterministically a path:
\[ \pi = u_0 u_1 \ldots u_n \ldots u_{n+m} \omega \] in \( TS \otimes G \)

where GNBA \( G_\varphi \) is obtained as explained in the previous lecture

2. check whether the guessed path is accepted by \( G_\varphi \)

NPSPACE Algorithm for \( \exists \text{LTL} \) MC Problem

1. Guess two natural number \( n \) and \( m \leq k \) such that \( m > 0 \) and
\[ k = |S| \cdot 2^{|cl(\varphi)|} \cdot |\varphi| \]

2. Guess initial path fragment \( \pi = s_0 s_1 \ldots s_n \ldots s_{n+m} \) in \( TS \)

3. Guess \( n+m+2 \) subsets \( B_0, \ldots, B_n, \ldots, B_{n+m+1} \) of \( cl(\varphi) \)

4. Check whether the following three conditions hold:
   1. \( \langle s_n, B_{n+1} \rangle = \langle s_{n+m}, B_{n+m+1} \rangle \)
   2. if \( \psi_1 \cup \psi_2 \in B_{n+1} \cup \ldots \cup B_{n+m} \) then \( \psi_2 \in B_{n+1} \cup \ldots \cup B_{n+m} \)
   3. \( n \leq |S| \cdot 2^{|cl(\varphi)|} \) and \( m \leq |S| \cdot 2^{|cl(\varphi)|} \cdot |\varphi| \)

If so, return “yes”, otherwise return “no”.

Criterion For \( \exists \text{LTL} \) Properties

There exists a path \( \pi \) in \( TS \) with \( \pi \models \varphi \) iff there exist:

1. an initial path fragment \( s_0 s_1 \ldots s_n \ldots s_{n+m} \) in \( TS \), and
2. a run \( B_0 B_1 \ldots B_{n+1} \ldots B_{n+m+1} \) in GNBA \( G_\varphi \) for trace(\( \pi \)) such that:
   1. \( \langle s_n, B_{n+1} \rangle = \langle s_{n+m}, B_{n+m+1} \rangle \)
   2. if \( \psi_1 \cup \psi_2 \in B_{n+1} \cup \ldots \cup B_{n+m} \) then \( \psi_2 \in B_{n+1} \cup \ldots \cup B_{n+m} \)
   3. \( n \leq |S| \cdot 2^{|cl(\varphi)|} \) and \( m \leq |S| \cdot 2^{|cl(\varphi)|} \cdot |\varphi| \)
LTL Model Checking is PSPACE Complete

The LTL model-checking problem is PSPACE-complete.

Proof.
We have seen that the $\exists$LTL model-checking problem is in NPSPACE = PSPACE. We will prove that this problem is PSPACE-hard. (This is done on the black board.) Since PSPACE = co-PSPACE, these two results together give that the LTL model-checking problem is PSPACE-complete.

We concentrate on PSPACE-hardness of the existential LTL model-checking problem.

LTL Satisfiability is PSPACE Complete

The LTL satisfiability problem is PSPACE-complete.

Proof.
The satisfiability of LTL-formula $\varphi$ amounts to checking whether $Words(\varphi) \neq \emptyset$. Automata-based satisfiability checking: construct the NBA $A_{\varphi}$. Check whether $L_\omega(A) \neq \emptyset$. This can be done using a nested DFS. The complexity is exponential in $|\varphi|$ . . . . . . . and PSPACE-complete.
Some Complexity Classes

Complexity of LTL Model Checking

Complexity of LTL Satisfiability

Summary

The LTL model-checking problem is PSPACE-complete

The LTL satisfiability problem is PSPACE-complete

The Hamiltonian path problem is polynomially reducible to the complement of the LTL model-checking problem.

Thursday November 21, 10:30