

Complexity and Correctness of LTL Model Checking

Lecture #16 of Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

E-mail: `katoen@cs.rwth-aachen.de`

June 5, 2007

Overview Lecture #16

⇒ Repetition: from LTL to GNBA

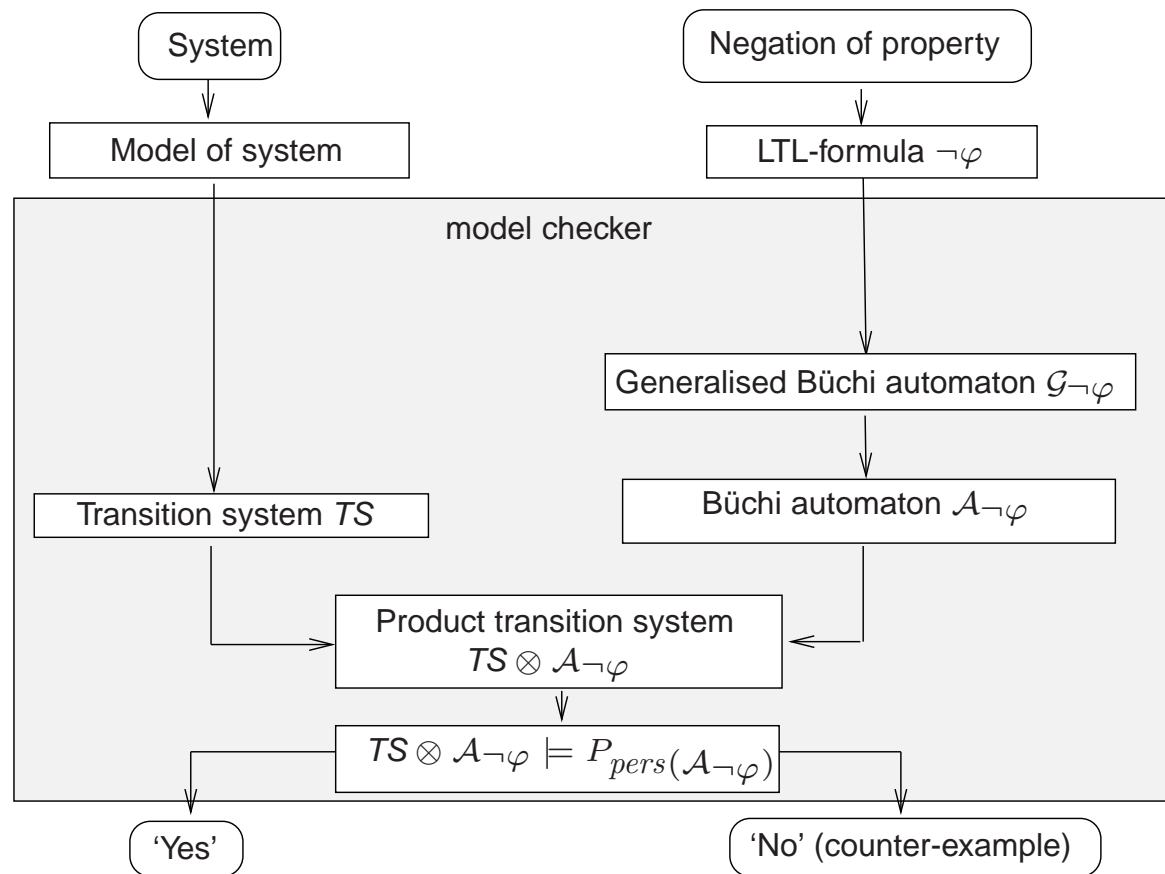
- Correctness proof
- Complexity results
 - LTL model checking is coNP-hard and PSPACE-complete
 - Satisfiability and validity are PSPACE-hard
- Summary of LTL model checking

Reduction to persistence checking

$$\begin{aligned} TS \models \varphi & \quad \text{if and only if} \quad \text{Traces}(TS) \subseteq \text{Words}(\varphi) \\ & \quad \text{if and only if} \quad \text{Traces}(TS) \cap ((2^{AP})^\omega \setminus \text{Words}(\varphi)) = \emptyset \\ & \quad \text{if and only if} \quad \text{Traces}(TS) \cap \underbrace{\text{Words}(\neg\varphi)}_{\mathcal{L}_\omega(\mathcal{A}_{\neg\varphi})} = \emptyset \\ & \quad \text{if and only if} \quad TS \otimes \mathcal{A}_{\neg\varphi} \models \Diamond\Box\neg F \end{aligned}$$

LTL model checking is thus reduced to persistence checking!

Overview of LTL model checking



From LTL to GNBA

GNBA \mathcal{G}_φ over 2^{AP} for LTL-formula φ with $\mathcal{L}_\omega(\mathcal{G}_\varphi) = \text{Words}(\varphi)$:

- Assume φ only contains the operators \wedge, \neg, \bigcirc and U
 - $\vee, \rightarrow, \diamond, \square, W$, and so on, are expressed in terms of these basic operators
- States are *elementary sets* of sub-formulas in φ
 - for $\sigma = A_0A_1A_2 \dots \in \text{Words}(\varphi)$, expand $A_i \subseteq AP$ with sub-formulas of φ
 - ... to obtain the infinite word $\bar{\sigma} = B_0B_1B_2 \dots$ such that

$$\psi \in B_i \quad \text{if and only if} \quad \sigma^i = A_iA_{i+1}A_{i+2} \dots \models \psi$$

- $\bar{\sigma}$ is intended to be a run in GNBA \mathcal{G}_φ for σ
- Transitions are derived from semantics \bigcirc and expansion law for U
- Accept sets guarantee that: $\bar{\sigma}$ is an accepting run for σ iff $\sigma \models \varphi$

Elementary sets of formulae

$B \subseteq \text{closure}(\varphi)$ is *elementary* if:

1. B is *logically consistent* if for all $\varphi_1 \wedge \varphi_2, \psi \in \text{closure}(\varphi)$:

- $\varphi_1 \wedge \varphi_2 \in B \Leftrightarrow \varphi_1 \in B \text{ and } \varphi_2 \in B$
- $\psi \in B \Rightarrow \neg\psi \notin B$
- $\text{true} \in \text{closure}(\varphi) \Rightarrow \text{true} \in B$

2. B is *locally consistent* if for all $\varphi_1 \cup \varphi_2 \in \text{closure}(\varphi)$:

- $\varphi_2 \in B \Rightarrow \varphi_1 \cup \varphi_2 \in B$
- $\varphi_1 \cup \varphi_2 \in B \text{ and } \varphi_2 \notin B \Rightarrow \varphi_1 \in B$

3. B is *maximal*, i.e., for all $\psi \in \text{closure}(\varphi)$:

- $\psi \notin B \Rightarrow \neg\psi \in B$

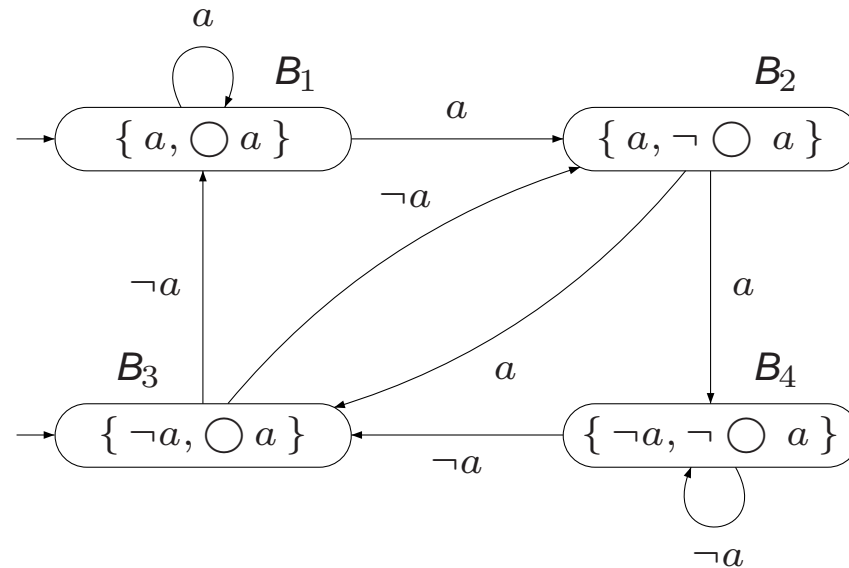
The GNBA of LTL-formula φ

For LTL-formula φ , let $\mathcal{G}_\varphi = (Q, 2^{AP}, \delta, Q_0, \mathcal{F})$ where

- Q = all elementary sets $B \subseteq \text{closure}(\varphi)$, $Q_0 = \{ B \in Q \mid \varphi \in B \}$
- $\mathcal{F} = \{ \{ B \in Q \mid \varphi_1 \cup \varphi_2 \notin B \text{ or } \varphi_2 \in B \} \mid \varphi_1 \cup \varphi_2 \in \text{closure}(\varphi) \}$
- The transition relation $\delta : Q \times 2^{AP} \rightarrow 2^Q$ is given by:
 - If $A \neq B \cap AP$ then $\delta(B, A) = \emptyset$
 - $\delta(B, B \cap AP)$ is the set of all elementary sets of formulas B' satisfying:
 - (i) For every $\bigcirc \psi \in \text{closure}(\varphi)$: $\bigcirc \psi \in B \Leftrightarrow \psi \in B'$, and
 - (ii) For every $\varphi_1 \cup \varphi_2 \in \text{closure}(\varphi)$:

$$\varphi_1 \cup \varphi_2 \in B \Leftrightarrow \left(\varphi_2 \in B \vee (\varphi_1 \in B \wedge \varphi_1 \cup \varphi_2 \in B') \right)$$

GNBA for LTL-formula $\bigcirc a$



$Q_0 = \{ B_1, B_3 \}$ since $\bigcirc a \in B_1$ and $\bigcirc a \in B_3$

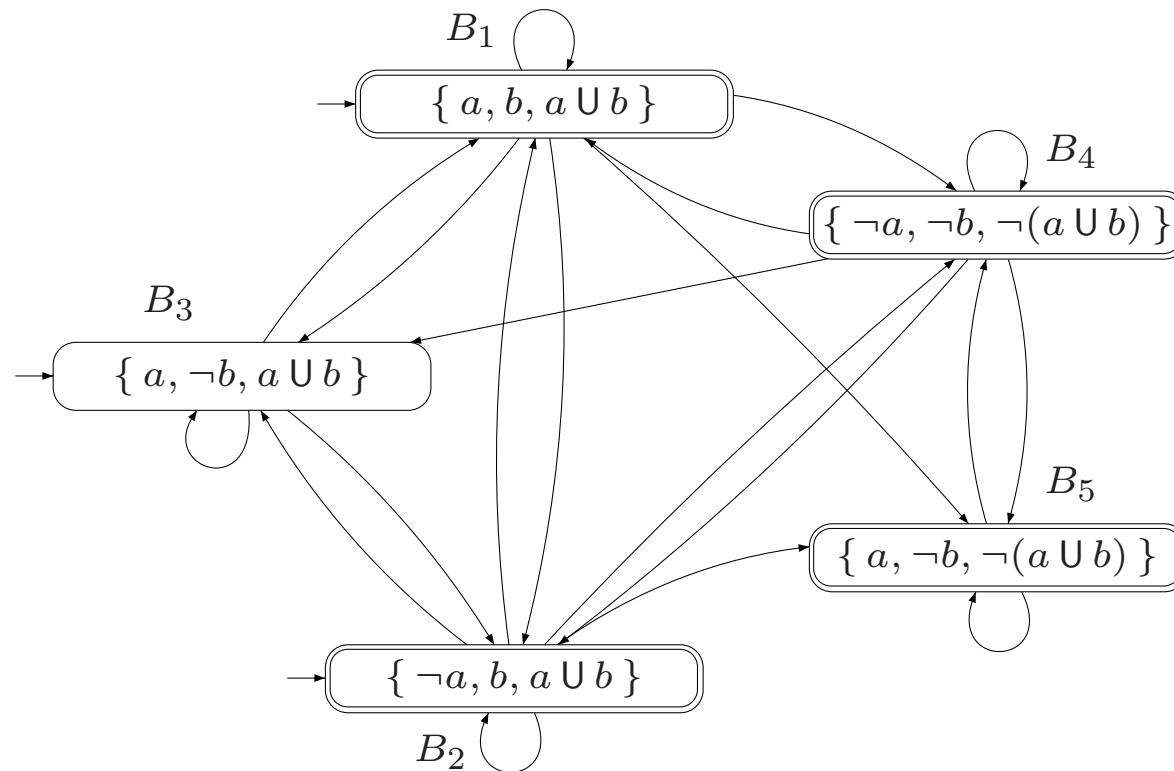
$\delta(B_2, \{ a \}) = \{ B_3, B_4 \}$ as $B_2 \cap \{ a \} = \{ a \}$, $\neg \bigcirc a = \bigcirc \neg a \in B_2$, and $\neg a \in B_3, B_4$

$\delta(B_1, \{ a \}) = \{ B_1, B_2 \}$ as $B_1 \cap \{ a \} = \{ a \}$, $\bigcirc a \in B_1$ and $a \in B_1, B_2$

$\delta(B_4, \{ a \}) = \emptyset$ since $B_4 \cap \{ a \} = \emptyset \neq \{ a \}$

The set \mathcal{F} is empty, since $\varphi = \bigcirc a$ does not contain an until-operator

GNBA for LTL-formula $a \cup b$



justification: on the black board

Overview Lecture #16

- Repetition: from LTL to GNBA

⇒ Correctness proof

- Complexity results
 - LTL model checking is coNP-hard and PSPACE-complete
 - Satisfiability and validity are PSPACE-hard
- Summary of LTL model checking

Correctness theorem

$$\text{Words}(\varphi) = \mathcal{L}_\omega(\mathcal{G}_\varphi)$$

Proof: on the black board

NBA are more expressive than LTL

Corollary: every LTL-formula expresses an ω -regular property

But: there exist ω -regular properties that cannot be expressed in LTL

Example: there is **no** LTL formula φ with $\text{Words}(\varphi) = P$ for the LT-property:

$$P = \left\{ A_0 A_1 A_2 \dots \in \left(2^{\{a\}} \right)^\omega \mid a \in A_{2i} \text{ for } i \geq 0 \right\}$$

But there exists an NBA \mathcal{A} with $\mathcal{L}_\omega(\mathcal{A}) = P$

\Rightarrow *there are ω -regular properties that cannot be expressed in LTL!*

Overview Lecture #16

- Repetition: from LTL to GNBA
- Correctness proof

⇒ Complexity results

- LTL model checking is coNP-hard and PSPACE-complete
- Satisfiability and validity are PSPACE-hard
- Summary of LTL model checking

Complexity for LTL to NBA

For any LTL-formula φ (over AP) there exists an NBA \mathcal{A}_φ
with $Words(\varphi) = \mathcal{L}_\omega(\mathcal{A}_\varphi)$ and
which can be constructed in time and space in $2^{\mathcal{O}(|\varphi| \cdot \log |\varphi|)}$

Justification complexity: next slide

Time and space complexity in $2^{\mathcal{O}(|\varphi| \cdot \log |\varphi|)}$

- States GNBA \mathcal{G}_φ are elementary sets of formulae in $\text{closure}(\varphi)$
 - sets B can be represented by bit vectors with single bit per subformula ψ of φ
- The number of states in \mathcal{G}_φ is bounded by $2^{|\text{subf}(\varphi)|}$
 - where $\text{subf}(\varphi)$ denotes the set of all subformulae of φ
 - $|\text{subf}(\varphi)| \leq 2 \cdot |\varphi|$; so, the number of states in \mathcal{G}_φ is bounded by $2^{\mathcal{O}(|\varphi|)}$
- The number of accepting sets of \mathcal{G}_φ is bounded above by $\mathcal{O}(|\varphi|)$
- The number of states in NBA \mathcal{A}_φ is thus bounded by $2^{\mathcal{O}(|\varphi|)} \cdot \mathcal{O}(|\varphi|)$
- $2^{\mathcal{O}(|\varphi|)} \cdot \mathcal{O}(|\varphi|) = 2^{\mathcal{O}(|\varphi| \log |\varphi|)}$

qed

Lower bound

There exists a family of LTL formulas φ_n with $|\varphi_n| = \mathcal{O}(\text{poly}(n))$
such that every NBA \mathcal{A}_{φ_n} for φ_n has at least 2^n states

Proof (1)

Let AP be non-empty, that is, $|2^{AP}| \geq 2$ and:

$$\mathcal{L}_n = \left\{ A_1 \dots A_n A_1 \dots A_n \sigma \mid A_i \subseteq AP \wedge \sigma \in \left(2^{AP}\right)^\omega \right\}, \quad \text{for } n \geq 0$$

It follows $\mathcal{L}_n = \text{Words}(\varphi_n)$ where $\varphi_n = \bigwedge_{a \in AP} \bigwedge_{0 \leq i < n} (\bigcirc^i a \longleftrightarrow \bigcirc^{n+i} a)$

φ_n is an LTL formula of polynomial length: $|\varphi_n| \in \mathcal{O}(|AP| \cdot n)$

However, any NBA \mathcal{A} with $\mathcal{L}_\omega(\mathcal{A}) = \mathcal{L}_n$ has at least 2^n states

Proof (2)

Claim: any NBA \mathcal{A} for $\bigwedge_{a \in AP} \bigwedge_{0 \leq i < n} (\bigcirc^i a \longleftrightarrow \bigcirc^{n+i} a)$ has at least 2^n states

Words of the form $A_1 \dots A_n A_1 \dots A_n \emptyset \emptyset \emptyset \dots$ are accepted by \mathcal{A}

\mathcal{A} thus has for every word $A_1 \dots A_n$ of length n , a state $q(A_1 \dots A_n)$, say, which can be reached from an initial state by consuming $A_1 \dots A_n$

From $q(A_1 \dots A_n)$, it is possible to visit an accept state infinitely often by accepting the suffix $A_1 \dots A_n \emptyset \emptyset \emptyset \dots$

If $A_1 \dots A_n \neq A'_1 \dots A'_n$ then

$$A_1 \dots A_n A'_1 \dots A'_n \emptyset \emptyset \emptyset \dots \notin \mathcal{L}_n = \mathcal{L}_\omega(\mathcal{A})$$

Therefore, the states $q(A_1 \dots A_n)$ are all pairwise different

Given $|2^{AP}|$ possible sequences $A_1 \dots A_n$, NBA \mathcal{A} has $\geq \left(|2^{AP}|\right)^n \geq 2^n$ states

Complexity for LTL model checking

The time and space complexity of LTL model checking is in $\mathcal{O}(|TS| \cdot 2^{|\varphi|})$

On-the-fly LTL model checking

- Idea: find a counter-example *during* the generation of $Reach(TS)$ and $\mathcal{A}_{\neg\varphi}$
 - exploit the fact that $Reach(TS)$ and $\mathcal{A}_{\neg\varphi}$ can be generated in parallel
- ⇒ Generate $Reach(TS \otimes \mathcal{A}_{\neg\varphi})$ “on demand”
- consider a new vertex only if no accepting cycle has been found yet
 - only consider the successors of a state in $\mathcal{A}_{\neg\varphi}$ that match current state in TS
- ⇒ Possible to find an accepting cycle *without generating $\mathcal{A}_{\neg\varphi}$ entirely*
- This *on-the-fly* scheme is adopted in e.g. the model checker SPIN

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

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

In fact, the LTL model-checking problem is PSPACE-complete

[Sistla & Clarke 1985]

LTL satisfiability and validity checking

- Satisfiability problem: $Words(\varphi) \neq \emptyset$ for LTL-formula φ ?
 - does there exist a transition system for which φ holds?
- Solution: construct an NBA \mathcal{A}_φ and check for emptiness
 - nested depth-first search for checking persistence properties
- Validity problem: is $\varphi \equiv \text{true}$, i.e., $Words(\varphi) = (2^{AP})^\omega$?
 - does φ hold for every transition system?
- Solution: as for satisfiability, as φ is valid iff $\neg\varphi$ is satisfiable

run time is exponential; a more efficient algorithm most probably does not exist!

LTL satisfiability and validity checking

The satisfiability and validity problem for LTL are PSPACE-complete

Black board: show the fact that these problems are PSPACE-hard

Overview Lecture #16

- Repetition: from LTL to GNBA
- Correctness proof
- Complexity results
 - LTL model checking is coNP-hard and PSPACE-complete
 - Satisfiability and validity are PSPACE-hard

⇒ Summary of LTL model checking

Summary of LTL model checking (1)

- LTL is a logic for formalizing **path**-based properties
- **Expansion law** allows for rewriting until into local conditions and next
- LTL-formula φ can be transformed algorithmically into NBA \mathcal{A}_φ
 - this may cause an exponential blow up
 - algorithm: first construct a GNBA for φ ; then transform it into an equivalent NBA
- LTL-formulae describe ω -regular LT-properties
 - but **do not have the same expressivity** as ω -regular languages

Summary of LTL model checking (2)

- $TS \models \varphi$ can be solved by a **nested depth-first search** in $TS \otimes \mathcal{A}_{\neg\varphi}$
 - time complexity of the LTL model-checking algorithm is linear in TS and exponential in $|\varphi|$
- Fairness assumptions can be described by LTL-formulae

the model-checking problem for LTL with fairness is reducible to the standard LTL model-checking problem
- **The LTL-model checking problem is PSPACE-complete**
- Satisfiability and validity of LTL amounts to NBA emptiness-check
- **The satisfiability and validity problem for LTL are PSPACE-complete**