

LTL Model Checking using Automata

Lecture #19 + #20 of Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

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

January 10, 2006

Overview Lecture #19 + #20

LTL and GNBA revisited

~~100%~~ From LTL to GNBA

~~100%~~ Complexity results

Linear Temporal Logic

modal logic over infinite sequences [Pnueli 1977]

Propositional logic

$\mathcal{L}_2 \ni a \quad AP$

$\mathcal{L}_2 \ni \neg$ and \wedge

atomic proposition
negation and conjunction

Temporal operators

$\mathcal{L}_2 \ni$

$\mathcal{L}_2 \ni U$

next state fulfills
holds Until a -state is reached

Auxiliary temporal operators

$\mathcal{L}_2 \ni$

true U

$\mathcal{L}_2 \ni$

$\mathcal{L}_2 \ni \mathcal{L}_2$

eventually
always

LTL model-checking problem

The following decision problem:

Given finite transition system TS and LTL-formula ϕ :
 yields ~~yes~~ if $TS \models \phi$, and ~~no~~ (plus a counterexample) if $TS \not\models \phi$

NBA for LTL-formulae

A first attempt

$$\begin{aligned}
 TS \models A & \quad \text{if and only if} \quad \text{Traces}(TS) \subseteq \overline{L(A)} \\
 & \quad \text{if and only if} \quad \text{Traces}(TS) \subseteq L(\overline{A}) =
 \end{aligned}$$

*but complementation of NBA is quadratically exponential
 if A has n states, \overline{A} has c^{n^2} states in worst case*

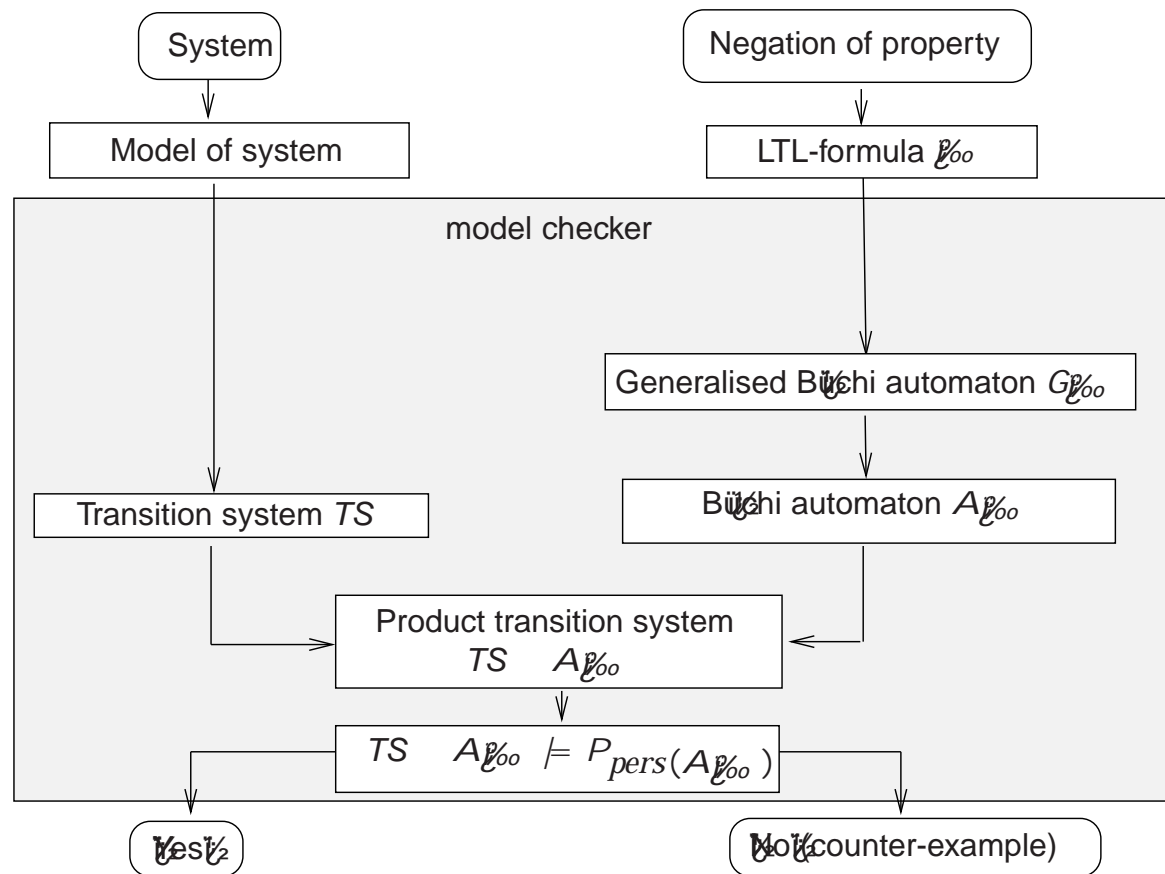
use the fact that $L(\overline{A}) = L(A_{\text{foo}})$!

Observation

$$\begin{aligned}
 TS \models & \quad \text{if and only if} \quad \text{Traces}(TS) \subseteq \text{Words}(\quad) \\
 & \quad \text{if and only if} \quad \text{Traces}(TS) \subseteq (2^{AP}) \setminus \text{Words}(\quad) = \\
 & \quad \text{if and only if} \quad \text{Traces}(TS) \subseteq \underbrace{\text{Words}(\mathcal{L}_{\text{oo}})}_{L(A_{\text{oo}})} = \\
 & \quad \text{if and only if} \quad TS \models A_{\text{oo}} \models \mathcal{L}_{\text{oo}}
 \end{aligned}$$

LTL model checking is thus reduced to persistence checking

Overview of LTL model checking



Generalized Büchi automata

A **generalized NBA** (GNBA) G is a tuple (Q, \rightarrow, Q_0, F) where:

Q is a finite set of states with $Q_0 \subseteq Q$ a set of initial states

Σ is an **alphabet**

$\rightarrow : Q \times \Sigma \rightarrow 2^Q$ is a **transition function**

$F = \{F_1, \dots, F_k\}$ is a (possibly empty) subset of 2^Q

The **size** of G , denoted $|G|$, is the number of states and transitions in G :

$$|G| = |Q| + \sum_{q \in Q} \sum_{A \in \Sigma} | \rightarrow(q, A) |$$

Language of a GNBA

GNBA $G = (Q, \delta, Q_0, F)$ and word $w = A_0A_1A_2 \dots$

A **run** for w in G is an infinite sequence $q_0 q_1 q_2 \dots$ such that:

$q_0 \in Q_0$ and $q_i \xrightarrow{A_i} q_{i+1}$ for all $0 \leq i$

Run $q_0 q_1 \dots$ is **accepting** if for all $F \in F$: $q_i \in F$ for infinitely many i

w is **accepted** by G if there exists an accepting run for w

The **accepted language** of G :

$$L(G) = \{ w \mid \text{there exists an accepting run for } w \text{ in } G \}$$

From GNBA to NBA

For any GNBA G there exists an NBA A with:

$$L(G) = L(A) \text{ and } |A| = O(|G| \cdot |F|)$$

where F denotes the set of acceptance sets in G

Overview Lecture #19 + #20

~~Code~~ LTL and GNBA revisited

From LTL to GNBA

~~Code~~ Complexity results

From LTL to GNBA

GNBA G over 2^{AP} for LTL-formula with $L(G) = \text{Words}(\)$:

States are *elementary sets* of sub-formulas in

for $\omega = A_0 A_1 A_2 \dots \in \text{Words}(\)$, expand A_i in AP with subformulas of ω to obtain the infinite word $\hat{\omega} = B_0 B_1 B_2 \dots$ such that

$$B_i \text{ is } \varphi \text{ if and only if } A_i A_{i+1} A_{i+2} \dots \models \varphi$$

subformulas of φ are considered as well as their *negation* $\neg \varphi$

Transitions are derived from semantics and expansion laws

Accepting conditions for G guarantee that:

ω is accepting if and only if $\hat{\omega} \models \varphi$

Closure

For LTL-formula φ , the set *closure*(φ) consists of all sub-formulas of φ and their negation $\neg\psi$ (where ψ and $\neg\psi$ are identified)

Elementary sets of formulae

B closure() is *elementary* if:

1. B is *logically consistent*:

$$\begin{aligned} & \not\models \varphi_1 \quad \varphi_2 \in B \quad \varphi_1 \in B \text{ and } \varphi_2 \in B \\ & \not\models B \quad \text{if } \varphi \in B \\ & \not\models \text{true} \quad \text{closure()} \quad \text{true} \in B \end{aligned}$$

2. B is *locally consistent*:

$$\begin{aligned} & \not\models \varphi_2 \in B \quad \varphi_1 \cup \varphi_2 \in B \\ & \not\models \varphi_1 \cup \varphi_2 \in B \text{ and } \varphi_2 \in B \quad \varphi_1 \in B \end{aligned}$$

3. B is *maximal*, i.e., for all closure():

$$\not\models \varphi \in B \quad \text{if } \varphi \in \text{closure}(B)$$

Examples

The GNBA of LTL-formula

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

Q is the set of all elementary sets of formulas $B \in \text{closure}(\phi)$

$$Q_0 = \{ B \in Q \mid B \in 2^{AP} \}$$

$$\rightarrow = \{ (B_1, B_2) \in Q \times Q \mid B_1 \cup B_2 \in \text{closure}(\phi) \}$$

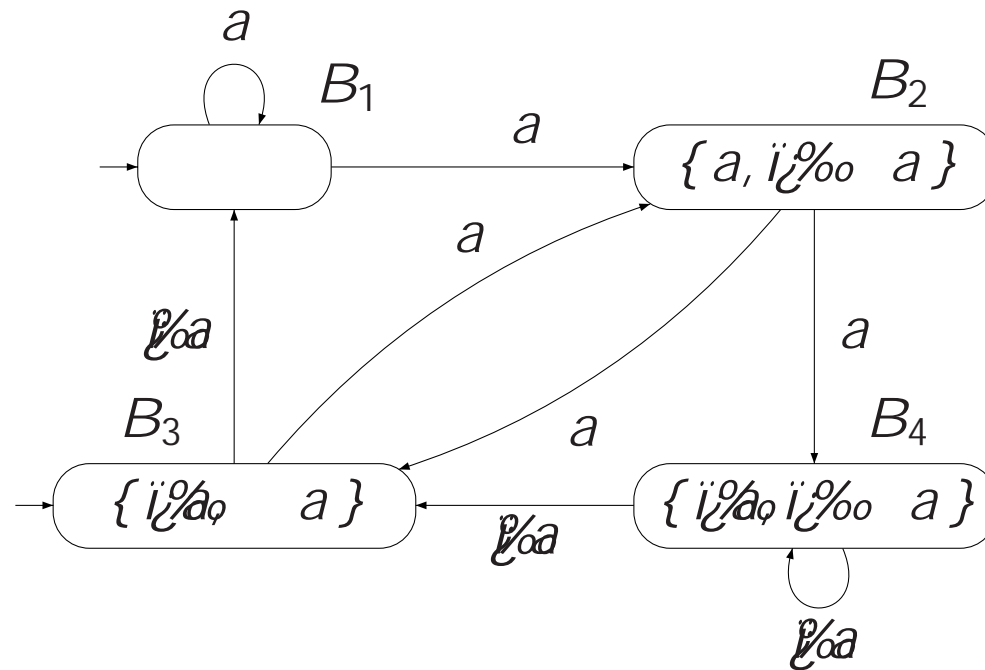
The transition relation $\rightarrow : Q \times Q$ is given by:

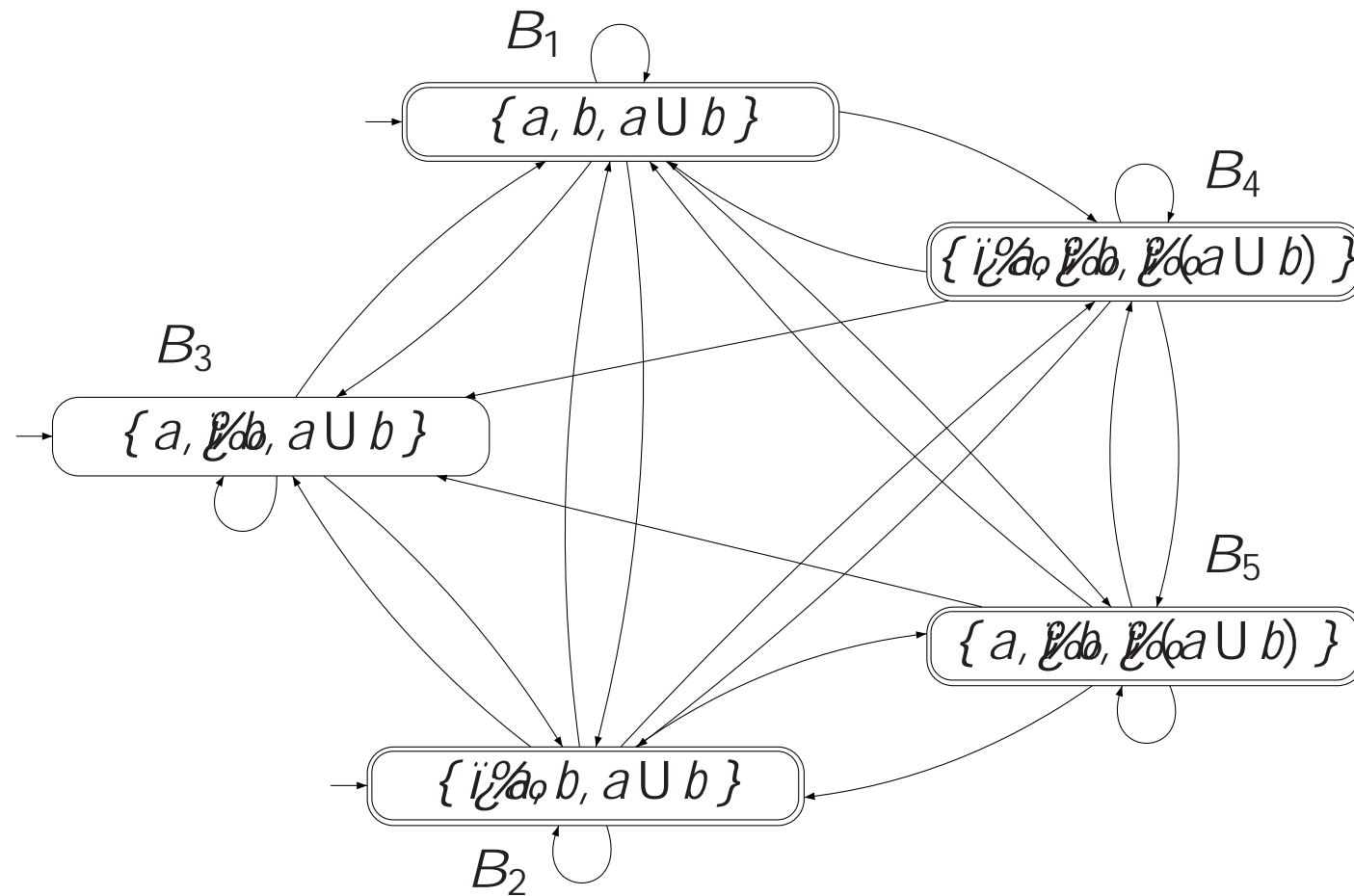
$(B, B') \in \rightarrow$ is the set of all elementary sets of formulas B satisfying:

- (i) For every $B' \in \text{closure}(\phi)$: $B \models B'$, and
- (ii) For every $B_1 \cup B_2 \in \text{closure}(\phi)$:

$$B \models B_1 \cup B_2 \iff B \models B_1 \text{ and } B \models B_2$$

GNBA for LTL-formula a



GNBA for LTL-formula $a \cup b$ 

Main result

[Vardi, Wolper & Sistla 1986]

For any LTL-formula φ (over AP) there exists a
GNBA G over 2^{AP} such that:

- (a) $Words(\varphi) = L(G)$
- (b) G can be constructed in time and space $O(|\varphi|^2)$
- (c) # accepting sets of G is bounded above by $O(|\varphi|)$

every LTL-formula expresses an ω -regular property!

Proof

NBA are more expressive than LTL

There is **no** LTL formula with $Words(\) = P$ for the LT-property:

$$P = A_0 A_1 A_2 \dots \quad 2^{\{a\}} \quad / a \quad A_{2i} \text{ for } i \geq 0$$

But there exists an NBA A with $L(A) = P$

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

Overview Lecture #19 + #20

~~Code~~ LTL and GNBA revisited

~~Code~~ From LTL to GNBA

Complexity results

Complexity for LTL to NBA

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

Lower bound

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

Complexity for LTL model checking

The time and space complexity of LTL model checking is in $O\left(\frac{|TS|}{\epsilon^2} \cdot \frac{1}{\epsilon^2}\right)$

On-the-fly LTL model checking

Idea: find a counter-example *during* the generation of $Reach(TS)$ and $A_{\neg\phi}$

½ exploit the fact that $Reach(TS)$ and $A_{\neg\phi}$ can be generated in parallel

Generate $Reach(TS \cap A_{\neg\phi})$ on demand ½

½ consider a new vertex only if no accepting cycle has been found yet

½ only consider the successors of a state in $A_{\neg\phi}$ that match current state in TS

Possible to find an accepting cycle *without generating $A_{\neg\phi}$ 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-hard

[Sistla & Clarke 1985]