

Model Checking

Abhik Roychoudhury
CS 5219
National University of Singapore

CS 5219 2007-08 by Abhik 1

Newsflash

- Turing Award 2007 (announced Feb 08)
 - Clarke, Emerson, Sifakis
 - For their role in developing Model-Checking into a highly effective verification technology, widely adopted in the hardware and software industries.

<http://awards.acm.org/homepage.cfm?prt=all&awd=140>

Award in 1996 for temporal logics to Amir Pnueli

CS 5219 2007-08 by Abhik 2

Model Checking

The flowchart shows the Model Checking (MC) process. It starts with 'Specification (e.g. Promela) Or Software' and 'Temporal Logic property e.g. in CTL or LTL'. Both lead to 'MC'. 'MC' also receives input from 'Finite State Transition Sys'. The output of 'MC' is either 'YES' or 'No, counterexample evidence (e.g. trace for LTL)'. A red note '??, in later lecture' points to the 'Specification' box.

CS 5219 2007-08 by Abhik 3

CTL Model Checking

- Inputs:
 - Finite state Kripke Structure $M = (S, I, \rightarrow, L)$
 - S (set of states),
 - $I \subseteq S$ (set of initial states),
 - \rightarrow (transition relation),
 - L (function labeling atomic propositions to states)
 - CTL formula φ (with atomic propositions corresponding to those appearing in M)
- Output:
 - Whether for all s in I , we have $M, s \models \varphi$

CS 5219 2007-08 by Abhik 4

Example: A microwave oven

The diagram shows a state transition system for a microwave oven. States are represented by circles with bit pairs (start, heat). Transitions are labeled with actions: start, open, close, done, cook, reset, warmup. A box at the bottom indicates the property $AG (start \Rightarrow AF heat)$.

Assignment of **start**, **heat** are shown

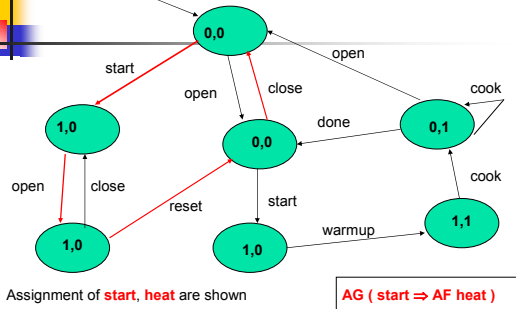
CS 5219 2007-08 by Abhik 5

Example

- $AG (start \Rightarrow AF heat)$
 - For any reachable state, if **start** holds, then along all outgoing paths, **heat** eventually holds.
 - Violated if:
 - \exists a reachable state s where **start** holds
 - \exists an acyclic path from s to s' in which **heat** does not hold in any state
 - And there is a cycle containing s' such that **heat** does not hold in all states of the cycle.
- Can we find such a cycle in the given model ?

CS 5219 2007-08 by Abhik 6

Example: A microwave oven



CS 5219 2007-08 by Abhik

7

CTL Model Checking

- A systematic way of doing the reasoning performed in our example for all CTL formulae.
- Recall set of all CTL formulae
- $\varphi = \text{Prop} \mid \neg \varphi \mid \varphi \wedge \varphi \mid \text{AX } \varphi \mid \text{EX } \varphi \mid \text{AF } \varphi \mid \text{EF } \varphi$
 $\mid \text{EG } \varphi \mid \text{AG } \varphi \mid \text{E}(\varphi \text{ U } \varphi) \mid \text{A}(\varphi \text{ U } \varphi)$
 $\mid \text{E}(\varphi \text{ R } \varphi) \mid \text{A}(\varphi \text{ R } \varphi)$
- Among the temporal operators, only consider
 - EX, EG, EU (along with \neg and \wedge)

CS 5219 2007-08 by Abhik

8

CTL temporal operators

- $\text{AX } \varphi = \neg \neg \text{AX } \varphi = \neg \text{EX } \neg \varphi$
- $\text{AG } \varphi = \neg \neg \text{AG } \varphi = \neg \text{EF } \neg \varphi$
- $\text{EF } \varphi = \text{E}(\text{true U } \varphi)$
- $\text{AF } \varphi = \neg \text{EG } \neg \varphi$
- $\text{A}(\varphi \text{ R } \Psi) = \neg \neg \text{A}(\varphi \text{ R } \Psi) = \neg \text{E}(\neg \varphi \text{ U } \neg \Psi)$
- $\text{A}(\varphi \text{ U } \Psi) = \neg \text{E}(\neg \varphi \text{ R } \neg \Psi)$
- What about **E($\varphi \text{ R } \Psi$)** ??

CS 5219 2007-08 by Abhik

9

Model Checking Algorithm

- Pre-processing: Rewrite the CTL formula to be verified to contain EX,EG,EU, \wedge , \neg
- For all sub-formulae x of the re-written formula φ , find the set of states satisfying x in the given model M .
- The above step finally computes the set of states in M which satisfy φ , call it $\text{St}(\varphi)$.
- Check whether all initial states of M are contained in $\text{St}(\varphi)$.

CS 5219 2007-08 by Abhik

10

Word of caution

- The pre-processing of the formula is only being done to simplify the presentation of the MC algorithm
 - Reduce # of cases to consider.
- We can
 - Develop customized algorithms for each of the 10 CTL operators, and then
 - apply them in a bottom-up recursive fashion as we will be doing now.

CS 5219 2007-08 by Abhik

11

Example

- M = the model of microwave oven given earlier
- $\varphi = \text{AG}(\text{start} \Rightarrow \text{AF heat})$
- $= \neg \text{EF}(\neg(\neg \text{start} \vee \text{AF heat}))$
- $= \neg \text{EF}(\text{start} \wedge \neg \text{AF heat})$
- $= \neg \text{EF}(\text{start} \wedge \text{EG } \neg \text{heat})$
- $= \neg \text{E}(\text{true U}(\text{start} \wedge \text{EG } \neg \text{heat}))$
- Now, how to compute the set of states in M satisfying this horrible formula ?

CS 5219 2007-08 by Abhik

12

Example

Essentially by bottom-up traversal of the formula

CS 5219 2007-08 by Abhik 13

Example

Steps in the traversal : Base case

CS 5219 2007-08 by Abhik 14

Example

Steps in the formula traversal

CS 5219 2007-08 by Abhik 15

Example

Steps in the formula traversal

CS 5219 2007-08 by Abhik 16

Example

Steps in the formula traversal

CS 5219 2007-08 by Abhik 17

Example

Steps in the formula traversal

CS 5219 2007-08 by Abhik 18

Example

Steps in the formula traversal

Check whether $St6$ contains all initial states of M

CS 5219 2007-08 by Abhik 19

Bottom-up formula traversal

Base case: Atomic propositions

The set of states satisfying p in $M = (S, S0, R, L)$, is found from L , the labeling function of M

Boolean operators: \wedge, \neg

CS 5219 2007-08 by Abhik 20

Questions Remaining

Temporal operators: EX, EU, EG

CS 5219 2007-08 by Abhik 21

EU

- Inputs:
 - Kripke Structure $M = (S, I, \rightarrow, L)$.
 - CTL formulae φ and Ψ .
 - $St(\varphi)$, set of states satisfying φ in M .
 - $St(\Psi)$, set of states satisfying Ψ in M .
- Output:
 - Set of states satisfying $E(\varphi \cup \Psi)$ in M .
- Technique:
 - Traversing the states (and transitions) of M .**

CS 5219 2007-08 by Abhik 22

$E(\varphi \cup \Psi)$: Intuition

CS 5219 2007-08 by Abhik 23

$E(\varphi \cup \Psi)$: Algorithm

- Result := $St(\Psi)$;
- Temp := $St(\Psi)$;
- while Temp \neq empty do
 - pick $s \in$ Temp; Temp := Temp - $\{s\}$;
 - Backstep := $\{s1 \mid s1 \rightarrow s, \text{ and } s1 \in St(\varphi)\}$;
 - Temp := Temp \cup Backstep;
 - Result := Result \cup Backstep;
- endwhile;
- return Result;

CS 5219 2007-08 by Abhik 24

EG

- Inputs:
 - Kripke Structure $M = (S, I, \rightarrow, L)$.
 - CTL formulae φ .
 - $St(\varphi)$, set of states satisfying φ in M .
- Output:
 - Set of states satisfying $EG \varphi$ in M .
- Technique:
 - Traversing the states (and transitions) of M .*

CS 5219 2007-08 by Abhik 25

EG φ : Intuition

The diagram shows a sequence of states represented by green circles. Arrows indicate transitions between them. A red path highlights a cycle of states that eventually leads to a state in $St(\varphi)$. The label $St(\varphi)$ is placed near the final state in the cycle.

CS 5219 2007-08 by Abhik 26

EG φ : Algorithm

- Result := $St(\varphi)$;
- repeat**
- Temp := { $s \mid s \in \text{Result}, \text{ and } \forall s1. s \rightarrow s1 \Rightarrow s1 \notin \text{Result} \}$;
- Result := Result - Temp;
- until** Temp = empty;
- return** Result;

CS 5219 2007-08 by Abhik 27

How to make it more efficient

- We initialize $St_{EGF} = St_f$
 - For each state in St_f , we check the out-edges. Many of the destination states are not in St_f , so cannot satisfy EGF
- It suffices to consider a reduced Kripke Structure M' constructed from M such that
 - All states of M which satisfy f are retained.
 - All other states and transitions are deleted.
- For any $s, M, s \models EG f$ if and only if
 - s is a state in M'
 - s reaches a state s' in M' where s' loops back to itself.

CS 5219 2007-08 by Abhik 28

Efficient computation

- Input: $M = (S, I, \rightarrow, L)$, St_f
- Output: St_{EGF}
- Technique:
 - Compute $M' = (S', I', \rightarrow', L')$ from M by keeping only nodes in St_f
 - Temp := St_{EGF} := All nodes in nontrivial SCCs of M'
 - while Temp \neq empty do
 - pick $s \in \text{Temp}$; Temp := Temp - { s };
 - $St_{EGF} := St_{EGF} \cup \{t \mid t \rightarrow' s \wedge t \notin St_{EGF}\}$;
 - Temp := Temp $\cup \{t \mid t \rightarrow' s \wedge t \notin St_{EGF}\}$;
 - endwhile

SCC = Strongly connected component = Maximal sub-graph where every node has a path to every other node.

CS 5219 2007-08 by Abhik 29

Summary, Exercises

- We have only presented model checking as a decision procedure.
- Other issues such as counter-example computation not shown.
- Direct iterative algorithms given only for EU, EG
 - What about EF, AF, AG etc. ?
- Algorithmic complexity of the iterative algorithms discussed in today's lecture.

CS 5219 2007-08 by Abhik 30