Introduction to Software Verification

Orna Grumberg

Lectures Material
winter 2015-16
Lecture 12

5.1.16
Several additional slides
Given an abstraction function $h : S \rightarrow S_h$, the concrete states are grouped and mapped into abstract states:
How to define an abstract model:

Given $M$ and $\varphi$, choose

- $S_h$ - a set of abstract states

- $AP$ - a set of atomic propositions that label concrete and abstract states

- $h : S \rightarrow S_h$ - a mapping from $S$ on $S_h$ that satisfies:

\[ h(s) = h(t) \text{ only if } L(s) = L(t) \]

- $h$ is called appropriate w.r.t. $AP$
The abstract model

\[ M_h = (S_h, I_h, R_h, L_h) \]

- \( s_h \in I_h \iff \exists s \in I : h(s) = s_h \)
- \( (s_h, t_h) \in R_h \iff \exists s, t \quad [ h(s) = s_h \land h(t) = t_h \land (s, t) \in R ] \)
- \( L_h(s_h) = L(s) \) for some \( s \) where \( h(s) = s_h \)

This is an exact abstraction
An approximated abstraction (an approximation)

• $s_h \in I_h \iff \exists s \in I : h(s) = s_h$

• $(s_h, t_h) \in R_h \iff$
  $\exists s, t \ [ h(s) = s_h \land h(t) = t_h \land (s, t) \in R ]$

• $L_h$ is as before

Notation:  
$M_r$ - reduced (exact)  
$M_h$ - approximated
Logic preservation Theorem

- **Theorem** If $\varphi$ is an ACTL/ACTL* specification over $AP$, then

$$M_h \models \varphi \Rightarrow M \models \varphi$$

$$(M_h \geq_{\text{sim}} M)$$

**Proof:** provide a simulation relation
End of lecture 12