Bisimulation equivalence: general idea

- M, M' bisimilar if they have 'corresponding executions'
  - to each step of M there is a corresponding step of M'
  - to each step of M' there is a corresponding step of M
- Bisimilar models satisfy same CTL* properties
- Bisimilar: same truth/falsity of model properties
- Simulation gives property-truth preserving abstraction (see later)

Bisimulation relations

- Let $R : S \to S \to \mathbb{B}$ and $R' : S' \to S' \to \mathbb{B}$ be transition relations
- B is a bisimulation relation between R and R' if:
  - $B : S \to S' \to \mathbb{B}$
  - $\forall s\ s'.\ B\ s\ s' \Rightarrow \forall s_1 \in S.\ R\ s\ s_1 \Rightarrow \exists s'_1.\ R'\ s'\ s'_1 \wedge B\ s_1\ s'_1$
    (to each step of R there is a corresponding step of R')
  - $\forall s\ s'.\ B\ s\ s' \Rightarrow \forall s'_1 \in S.\ R'\ s'\ s'_1 \Rightarrow \exists s_1.\ R'\ s\ s_1 \wedge B\ s_1\ s'_1$
    (to each step of R' there is a corresponding step of R)

Bisimulation equivalence: definition and theorem

- Let $M = (S, S_0, R, L)$ and $M' = (S', S'_0, R', L')$
- $M \equiv M'$ if:
  - there is a bisimulation B between R and R'
  - $\forall s_0 \in S_0.\ \exists s'_0 \in S'_0.\ B\ s_0\ s'_0$
  - $\forall s'_0 \in S'_0.\ \exists s_0 \in S_0.\ B\ s_0\ s'_0$
  - there is a bijection $\theta : AP \to AP'$
  - $\forall s\ s'.\ B\ s\ s' \Rightarrow L(s) = L'(s')$
- Theorem: if $M \equiv M'$ then for any CTL* state formula ψ:
  $M \models \psi \ \Leftrightarrow\ M' \models \psi$
- See Q14 in the Exercises

Abstraction

- Abstraction creates a simplification of a model
  - separate states may get merged
  - an abstract path can represent several concrete paths
- $M \preceq \overline{M}$ means $\overline{M}$ is an abstraction of M
  - to each step of M there is a corresponding step of $\overline{M}$
  - atomic properties of M correspond to atomic properties of $\overline{M}$
- Special case is when $\overline{M}$ is a subset of M such that:
  - $\overline{M} = (\overline{S_0}, \overline{S}, \overline{R}, \overline{L})$ and $M = (S_0, S, R, L)$
    $\overline{S} \subseteq S$
    $\overline{S_0} = S_0$
    $\forall s\ s' \in \overline{S}.\ \overline{R}\ s\ s' \Leftrightarrow R\ s\ s'$
    $\forall s \in \overline{S}.\ \overline{L}\ s = L\ s$
  - $\overline{S}$ contain all reachable states of M
    $\forall s \in \overline{S}.\ \forall s' \in S.\ R\ s\ s' \Rightarrow s' \in \overline{S}$
- All paths of M from initial states are $\overline{M}$-paths
  - hence for all CTL formulas ψ: $\overline{M} \models \psi \ \Rightarrow M \models \psi$

Recall JM1

Thread 1	Thread 2
`0:  IF LOCK=0 THEN LOCK:=1;`	`0:  IF LOCK=0 THEN LOCK:=1;`
`1:  X:=1;`	`1:  X:=2;`
`2:  IF LOCK=1 THEN LOCK:=0;`	`2:  IF LOCK=1 THEN LOCK:=0;`
`3:`	`3:`

- Two program counters, state: $(pc_1, pc_2, lock, x)$

$$S_{\texttt{JM1}} = [0..3] \times [0..3] \times \mathbb{Z} \times \mathbb{Z}$$

$$\begin{array}{lll|lll} R_{\texttt{JM1}} & (0, pc_2, 0, x) & (1, pc_2, 1, x) & R_{\texttt{JM1}} & (pc_1, 0, 0, x) & (pc_1, 1, 1, x) \\ R_{\texttt{JM1}} & (1, pc_2, lock, x) & (2, pc_2, lock, 1) & R_{\texttt{JM1}} & (pc_1, 1, lock, x) & (pc_1, 2, lock, 2) \\ R_{\texttt{JM1}} & (2, pc_2, 1, x) & (3, pc_2, 0, x) & R_{\texttt{JM1}} & (pc_1, 2, 1, x) & (pc_1, 3, 0, x) \end{array}$$

- Assume $\texttt{NotAt11} \in L_{\texttt{JM1}}(pc_1, pc_2, lock, x) \Leftrightarrow \neg((pc_1 = 1) \wedge (pc_2 = 1))$
- Model $M_{\texttt{JM1}} = (S_{\texttt{JM1}}, \{(0,0,0,0)\}, R_{\texttt{JM1}}, L_{\texttt{JM1}})$
- $S_{\texttt{JM1}}$ not finite, but actually $lock \in \{0,1\}$, $x \in \{0,1,2\}$
- Clear by inspection that $M_{\texttt{JM1}} \preceq \overline{M}_{\texttt{JM1}}$ where:

$$\overline{M}_{\texttt{JM1}} = (\overline{S}_{\texttt{JM1}}, \{(0,0,0,0)\}, \overline{R}_{\texttt{JM1}}, \overline{L}_{\texttt{JM1}})$$

  - $\overline{S}_{\texttt{JM1}} = [0..3] \times [0..3] \times [0..1] \times [0..3]$
  - $\overline{R}_{\texttt{JM1}}$ is $R_{\texttt{JM1}}$ restricted to arguments from $\overline{S}_{\texttt{JM1}}$
  - $\texttt{NotAt11} \in \overline{L}_{\texttt{JM1}}(pc_1, pc_2, lock, x) \Leftrightarrow \neg((pc_1 = 1) \wedge (pc_2 = 1))$
  - $\overline{L}_{\texttt{JM1}}$ is $L_{\texttt{JM1}}$ restricted to arguments from $\overline{S}_{\texttt{JM1}}$

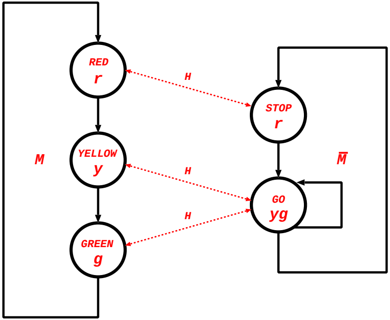
Simulation relations

- Let $R : S \to S \to \mathbb{B}$ and $\overline{R} : \overline{S} \to \overline{S} \to \mathbb{B}$ be transition relations
- H is a simulation relation between R and $\overline{R}$ if:
  - H is a relation between S and $\overline{S}$ – i.e. $H : S \to \overline{S} \to \mathbb{B}$
  - to each step of R there is a corresponding step of $\overline{R}$ – i.e.:
    $\forall s\ \overline{s}.\ H\ s\ \overline{s} \Rightarrow \forall s' \in S.\ R\ s\ s' \Rightarrow \exists \overline{s'} \in \overline{S}.\ \overline{R}\ \overline{s}\ \overline{s'} \wedge H\ s'\ \overline{s'}$
- Also need to consider abstraction of atomic properties
  - $H_{AP} : AP \to \overline{AP} \to \mathbb{B}$
  - details glossed over here

Simulation preorder: definition and theorem

- Let $M = (S, S_0, R, L)$ and $\overline{M} = (\overline{S}, \overline{S_0}, \overline{R}, \overline{L})$
- $M \preceq \overline{M}$ if:
  - there is a simulation H between R and $\overline{R}$
  - $\forall s_0 \in S_0.\ \exists \overline{s_0} \in \overline{S_0}.\ H\ s_0\ \overline{s_0}$
  - $\forall s\ \overline{s}.\ H\ s\ \overline{s} \Rightarrow L(s) = \overline{L}(\overline{s})$
- ACTL is the subset of CTL without **E**-properties
  - e.g. **AG AF**p – from anywhere can always reach a p-state
- Theorem: if $M \preceq \overline{M}$ then for any ACTL state formula ψ: $\overline{M} \models \psi \Rightarrow M \models \psi$
- If $\overline{M} \models \psi$ fails then cannot conclude $M \models \psi$ false

Example (Grumberg)



H a simulation

H *RED STOP* $\wedge$
H *YELLOW GO* $\wedge$
H *GREEN GO*

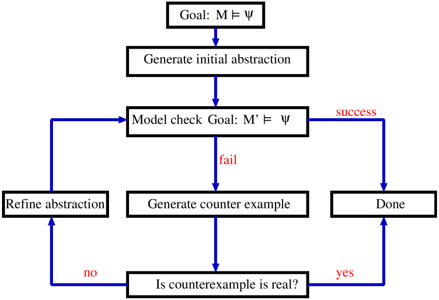
$H_{AP} : \{r, y, g\} \to \{r, yg\} \to \mathbb{B}$

$H_{AP}\ r\ r \wedge$
$H_{AP}\ y\ yg \wedge$
$H_{AP}\ g\ yg$

- $\overline{M} \models$ **AG AF** $\neg r$ hence $M \models$ **AG AF** $\neg r$
- but $\neg(\overline{M} \models$ **AG AF** $r)$ doesn't entail $\neg(M \models$ **AG AF** $r)$
  - $[\![$**AG AF** $r]\!]_{\overline{M}}(STOP)$ is false
    (consider $\overline{M}$-path π' where $\pi' = STOP.GO.GO.GO.\cdots$)
  - $[\![$**AG AF** $r]\!]_{M}(RED)$ is true
    (abstract path π' doesn't correspond to a real path in M)

CEGAR

- Counter Example Guided Abstraction Refinement



- Lots of details to fill out (several different solutions)
  - how to generate abstraction
  - how to check counterexamples
  - how to refine abstractions
- Microsoft SLAM driver verifier is a CEGAR system

Temporal Logic and Model Checking – Summary

- Various property languages: LTL, CTL, PSL (Prior, Pnueli)
- Models abstracted from hardware or software designs
- Model checking checks $M \models \psi$ (Clarke et al.)
- Symbolic model checking uses BDDs (McMillan)
- Avoid state explosion via simulation and abstraction
- CEGAR refines abstractions by analysing counterexamples
- Triumph of application of computer science theory
  - two Turing awards, McMillan gets 2010 CAV award
  - widespread applications in industry

Topics and corresponding slides

Topic	*Slides*
Introduction to models	1 - 9
Atomic properties	10
Trees and paths	11 - 12
Examples of properties	13 - 16
Reachability	17
Introduction to model checking	18 - 26
Symbolic model checking	27 - 32
Disjunctive partitioning of BDDs	33 - 35
Generating counter-examples	36 - 42
Introduction to temporal logic	43 - 45
Linear Temporal Logic (LTL)	46 - 58
Computation Tree Logic (CTL)	59 - 75
CTL model checking	75 - 83
History of model checking	84
Expressibility of LTL and CTL	57 - 58, 85 - 87
CTL*	88 - 90
Fairness	91 - 92
Propositional modal μ-calculus	93
Sequential Extended Regular Expressions (SEREs)	94 - 95
Assertion Based Verification (ABV) and PSL	96 - 107
Dynamic verification: event semantics	108 - 117
Bisimulation	118 - 120
Abstraction	121 - 125
Counterexample Guided Abstraction Refinement (CEGAR)	126
Summary	127

THE END