Lecture 16 - March 16

Reactive System: Bridge Controller

Announcements

- ProgTest1 result to be released by the end of Friday
- Lab3 released
- Example Questions for Written Test 2 released
  review session.
- To be completed by the final exam:
  Makeup lectures for WT1, WT2, ProgTest1, ProgTest2

Lecture

Reactive System: Bridge Controller

First Refinement: Invariant Preservation
Concrete, Refined Events

"old" events, existing in both m0 and m1.

PO/VC Rule of Invariant Preservation: Sequents

** $\cancel{a}=0 \vee \cancel{c}=0$
a $\quad$ $c-1$
$\equiv$
$a=0 \vee c-1=0$

Abstract m0

variables: n

invariants: I
inv0_1 : $n \in \mathbb{N}$
inv0_2 : $n \leq d$

ML_out
when
$n < d$
then
$n := n + 1$ $\quad n' = n+1$
end

ML_in
when
$n > 0$
then
$n := n - 1$
end

$A(c)$ → axioms
$I(c, v)$ → abstract inv.
$J(c, v, w)$ → concrete inv.
$H(c, w)$ concrete guard.
$\vdash$
$J_i(c, E(c, v), F(c, w))$

i: 1..5
$E(c, v)$: effect of e in the abs. state
$F(c, w)$: effect of e in con. state

* $\cancel{a} + \cancel{b} + \cancel{c} = \cancel{n}$
$\quad a+1 \quad b \quad c \quad n+1$
$(a+1)+b+c = (n+1)$

Concrete m1

variables: a, b, c

invariants: J
inv1_1 : $a \in \mathbb{N}$
inv1_2 : $b \in \mathbb{N}$
inv1_3 : $c \in \mathbb{N}$
inv1_4 : $a + b + c = n$
inv1_5 : $a = 0 \vee c = 0$

ML_out
when H
$a + b < d$
$c = 0$
then
$a := a + 1$
end

BAP: $a' = a+1 \wedge b' = b \wedge c' = c$

ML_in
when
$c > 0$ H
then
$c := c - 1$
end

BAP: $c' = c-1 \wedge a' = a \wedge b' = b$

ML-out, ML-in → 5 inv. in m1
10 (2 * 5)

Q. How many PO/VC rules for model m1?

ML_out/inv1_4/INV

$d \in \mathbb{N}$	axm0_1
$d > 0$	axm0_2
$n \in \mathbb{N}$	inv0_1
$n \leq d$	inv0_2

$a \in \mathbb{N}$
$b \in \mathbb{N}$
$c \in \mathbb{N}$
$a+b+c = n$
$a = 0 \vee c = 0$
$a+b<d$] ML_out grds
$c = 0$
$\vdash$ *

ML_in/inv1_5/INV

$d \in \mathbb{N}$	axm0_1
$d > 0$	axm0_2
$n \in \mathbb{N}$	inv0_1
$n \leq d$	inv0_2

$a \in \mathbb{N}$
$b \in \mathbb{N}$
$c \in \mathbb{N}$
$a+b+c = n$
$a = 0 \vee c = 0$
$c > 0$] grd of ML_in
$\vdash$ **

Commuting diagram.

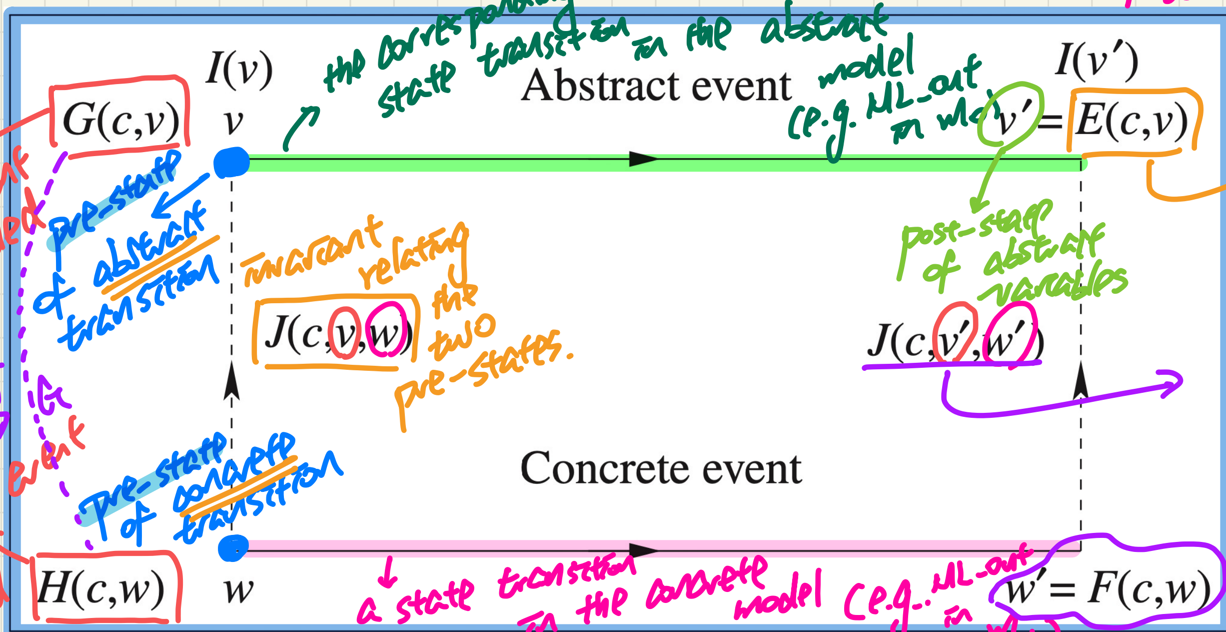
Visualizing Invariant Preservation in Refinement

Each **concrete state transition** (from w to w′)
should be simulated by
an **abstract state transition** (from v to v′)

$w' = F(c, w)$

effect of concrete transition

post-state of concrete variables



effect of abs. transition

same invariant holds at the post-state