Exam 2014
(partial solution – part III)

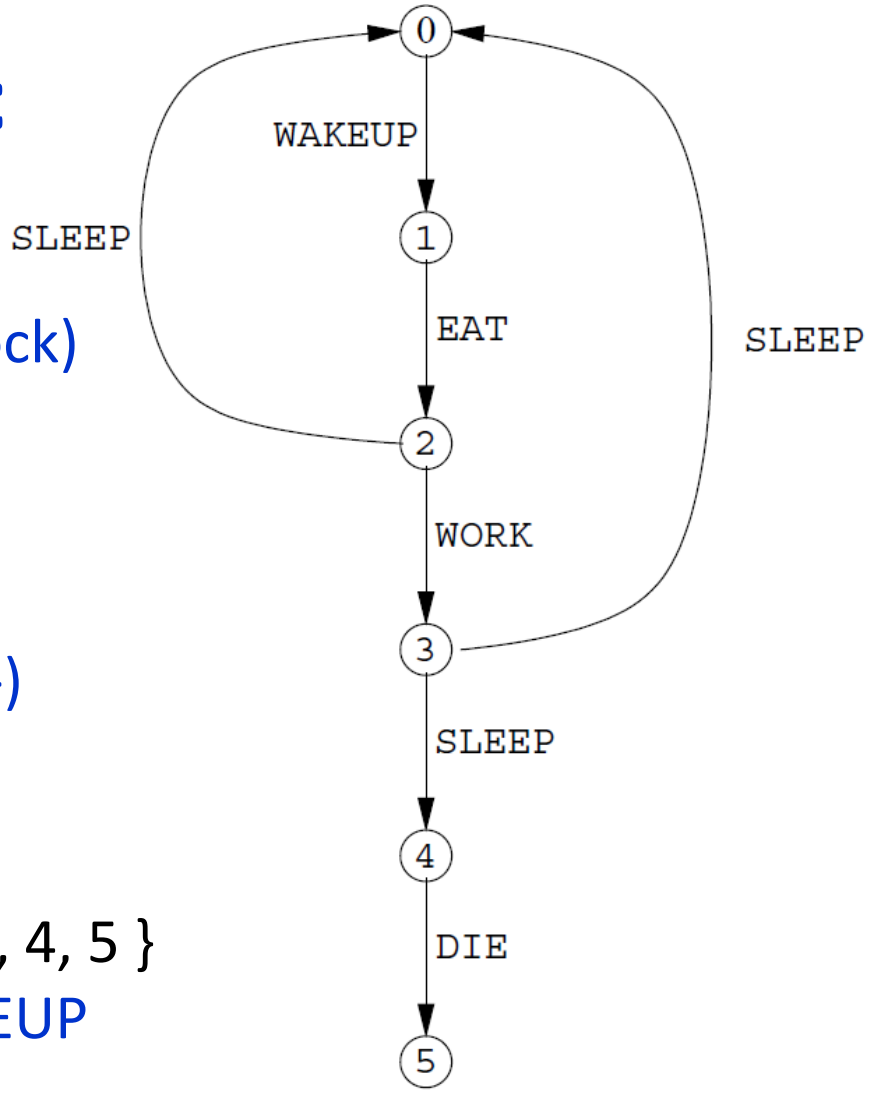
Radu Mateescu

Inria and LIG / Convecs

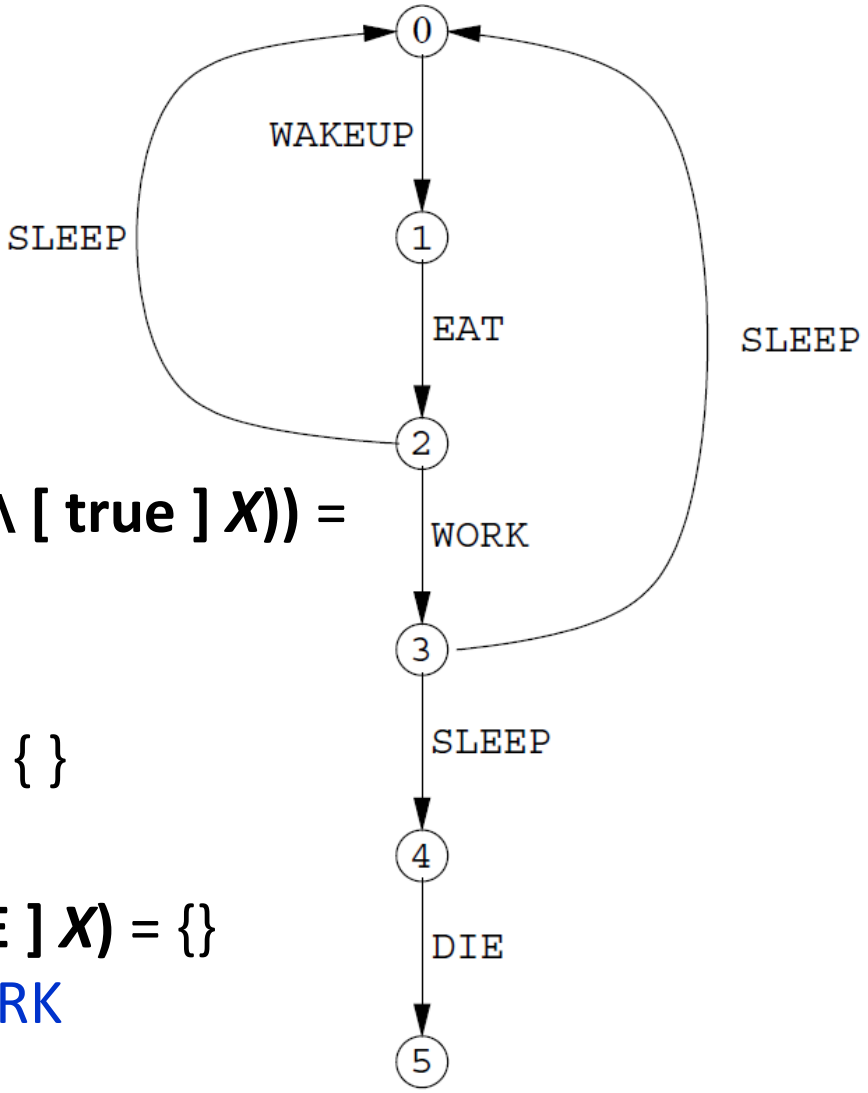
http://convecs.inria.fr

Part III: Temporal logic

1. ⟨ **true** ⟩ **true** = { 0, 1, 2, 3, 4 }
   // at least one successor (no deadlock)
2. ⟨ **EAT** ⟩ **true** = { 1 }
   // may immediately EAT
3. **[WAKEUP] false** = { 1, 2, 3, 4, 5 }
   // no immediate WAKEUP (please :-)
4. **[WORK] true** = { 0, …, 5 }
   // tautology!
5. **[SLEEP]** ⟨ **WAKEUP** ⟩ **true** = { 0, 1, 2, 4, 5 }
   // after each SLEEP, there is a WAKEUP



Part III: Temporal logic



6. $\mu X . (\langle \text{WORK} \rangle \text{true} \vee \langle \text{true} \rangle X) =$
   $\{ 0, 1, 2, 3 \}$
   // potentially WORK
7. $\mu X . (\langle \text{WORK} \rangle \text{true} \vee (\langle \text{true} \rangle \text{true} \wedge [\text{true}] X)) =$
   $\{ 0, 1, 2 \}$
   // inevitable *reachability* of WORK
8. $\mu X . (\langle \text{true} \rangle \text{true} \wedge [\neg \text{WORK}] X) = \{ \}$
   // inevitable *execution* of WORK
9. $\nu X . (\langle \text{true}^* . \text{WORK} \rangle \text{true} \wedge [\neg \text{DIE}] X) = \{\}$
   // as long as no DIE, potentially WORK
10. $\nu X. \langle \neg \text{WORK} \rangle X = \{ 0, 1, 2 \}$
    // may not WORK forever

Part III: Temporal logic

- Iterative computation (minimal fixed point):

$$\varphi_7 = \mu X . (\langle \mathbf{WORK} \rangle \mathbf{true} \vee (\langle \mathbf{true} \rangle \mathbf{true} \wedge [\mathbf{true}] X))$$

$X_0 = \{ \}$ // empty set

$X_1 = [[\langle \text{WORK} \rangle \text{true} \vee (\langle \text{true} \rangle \text{true} \wedge [\text{true}] X)]] [\{ \} / X] = \{ 2 \}$

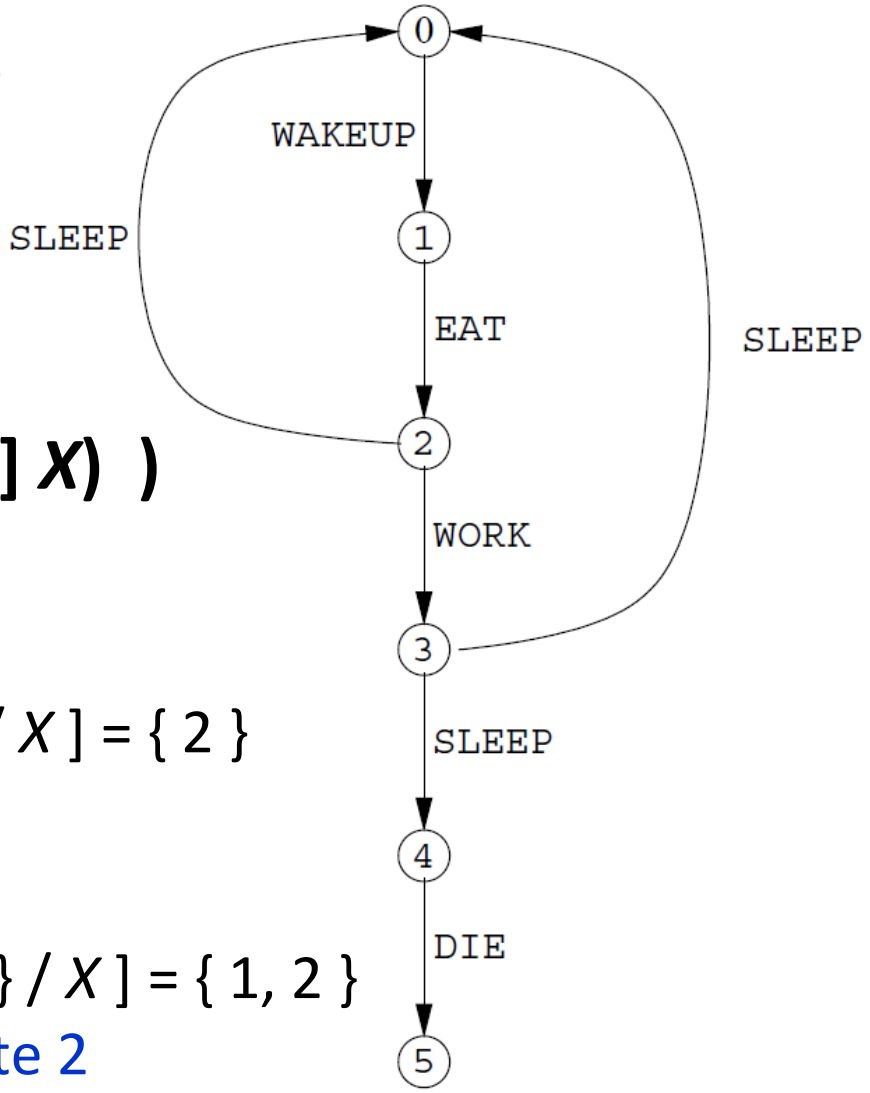
// do some WORK

$X_2 = [[\langle \text{WORK} \rangle \text{true} \vee (\langle \text{true} \rangle \text{true} \wedge [\text{true}] X)]] [\{ 2 \} / X] = \{ 1, 2 \}$

// do some WORK, or go to state 2

$X_3 = [[\langle \text{WORK} \rangle \text{true} \vee (\langle \text{true} \rangle \text{true} \wedge [\text{true}] X)]] [\{1,2\} / X] = \{ 0, 1, 2 \}$

// do some WORK, or go to states 1 or 2 (stabilization)



Part III: Temporal logic

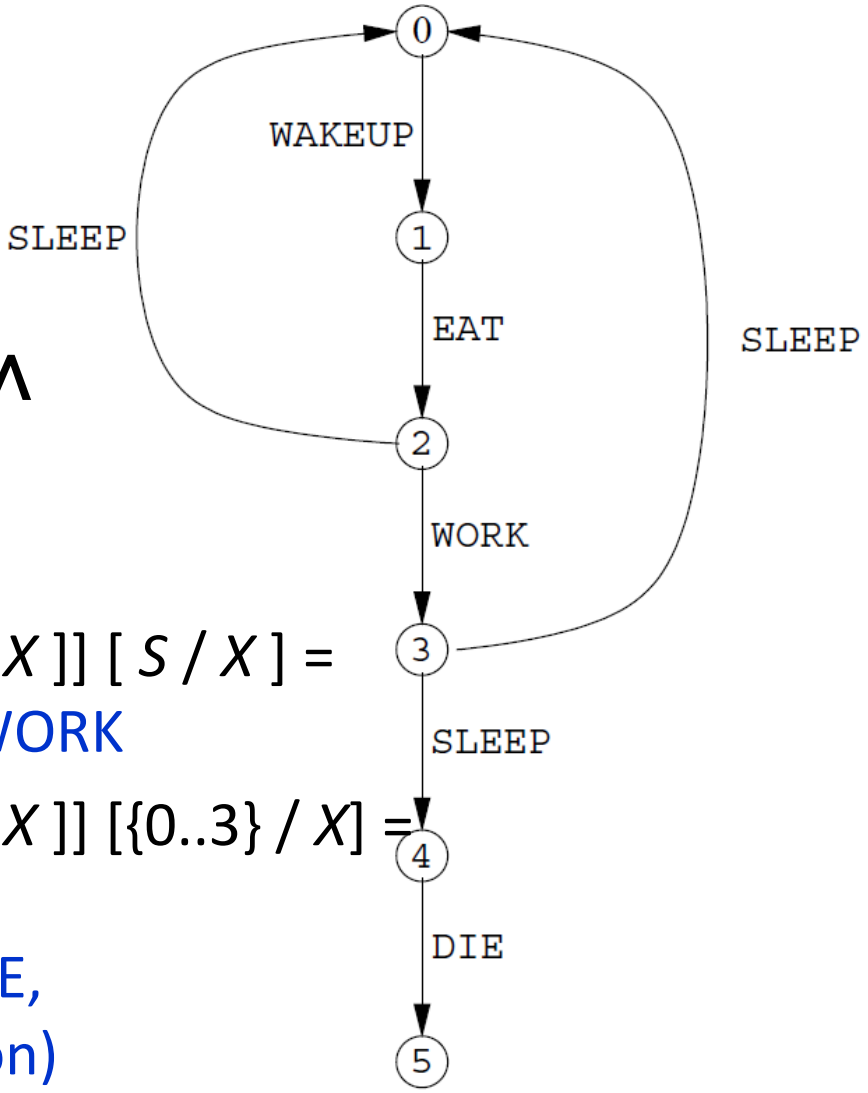
- Iterative computation (maximal fixed point):

$$\varphi_9 = \nu X \,.\, (\ \langle\, \mathbf{true^*.\ WORK}\ \rangle\, \mathbf{true} \wedge [\ \neg \mathbf{DIE}\ ]\ X\ )$$

$X_0 = \{ 0, \ldots, 5 \}$ // full set

$X_1 = [[\ \langle\, \text{true}^*.\ \text{WORK}\ \rangle\, \text{true} \wedge [\ \neg \text{DIE}\ ]\ X\ ]]\ [\ S\ /\ X\ ] =$
$\{ 0, 1, 2, 3 \}$ // reach some WORK

$X_2 = [[\ \langle\, \text{true}^*.\ \text{WORK}\ \rangle\, \text{true} \wedge [\ \neg \text{DIE}\ ]\ X\ ]]\ [\{0..3\}\ /\ X] =$
$\{ 0, 1, 2, 3 \}$
// do some WORK and if not DIE,
// go to a state 0..3 (stabilization)



- In PDL: $\varphi_9 = \mathbf{[\ (\neg DIE)^*\ ]\ \langle\ true^*.\ WORK\ \rangle\ true}$
(see the encodings of PDL iteration modalities in mu-calculus)

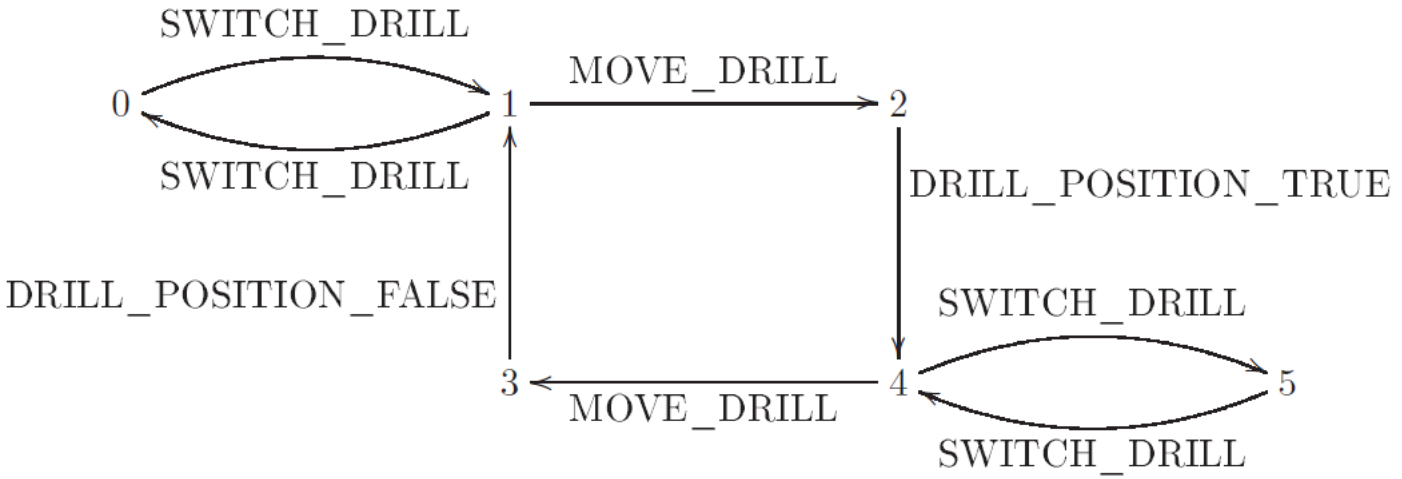
Exam 2017
(partial solution – part II)

Radu Mateescu

Inria and LIG / Convecs

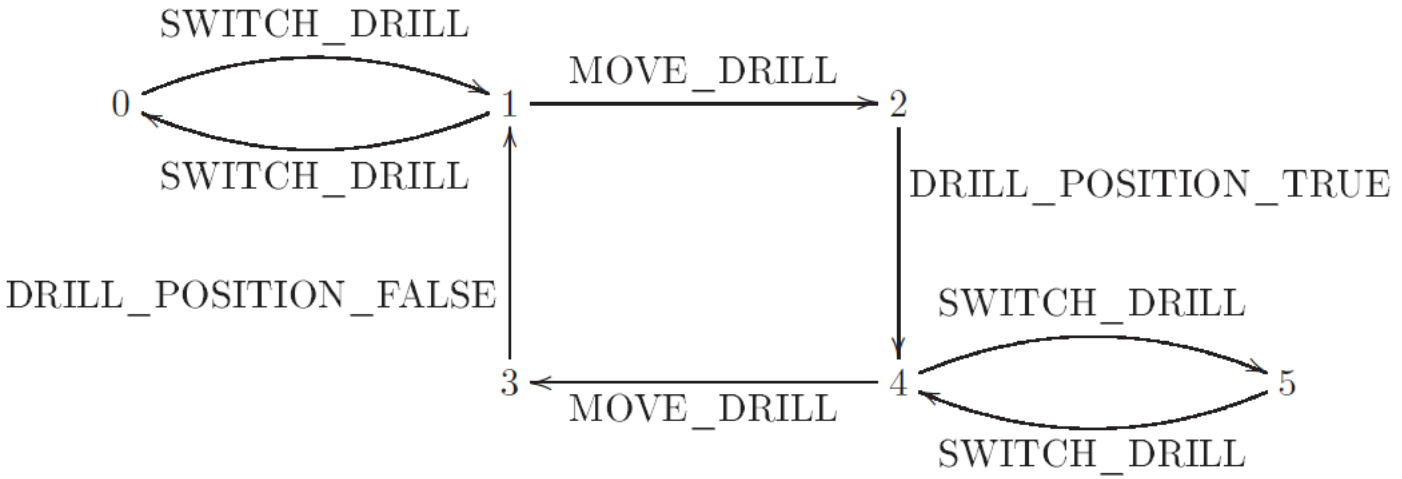
http://convecs.inria.fr

Part II: Temporal logic



1. [**true**] **false** = { } // deadlock
2. ⟨ **MOVE_DRILL** ⟩ **true** = { 1, 4 } // may immediately move the drill
3. **[SWITCH_DRILL] false** = { 1, 2, 3, 4, 5 } // cannot switch the drill
4. **[DRILL_POSITION_TRUE] true** = { 0, 1, 2, 3, 4, 5 } // tautology!
5. ⟨ **true*. DRILL_POSITION_TRUE** ⟩ **true** = { 0, 1, 2, 3, 4, 5 }
   // potentially put the drill in position true
6. **[MOVE_DRILL . (¬DRILL_POSITION_FALSE)*. MOVE_DRILL] false**
   = { 0, 2, 3, 4, 5 }
   // cannot move the drill twice without putting it in position false

Part II: Temporal logic



7. μX . ⟨ **DRILL_POSITION_TRUE** ⟩ **true** $\vee$ ⟨ **DRILL_POSITION_FALSE** ⟩ **true** $\vee$ ⟨ **MOVE_DRILL** ⟩ X = { 1, 2, 3, 4 }
   // potentially put the drill in position true or false
8. νX . ⟨ **SWITCH_DRILL** ⟩ X = { 0, 1, 4, 5 }
   // may infinitely switch the drill

- Subsidiary question: write formula 7 in PDL

  ⟨ **MOVE_DRILL***. **DRILL_POSITION_TRUE** $\vee$ **DRILL_POSITION_FALSE** ⟩ **true**

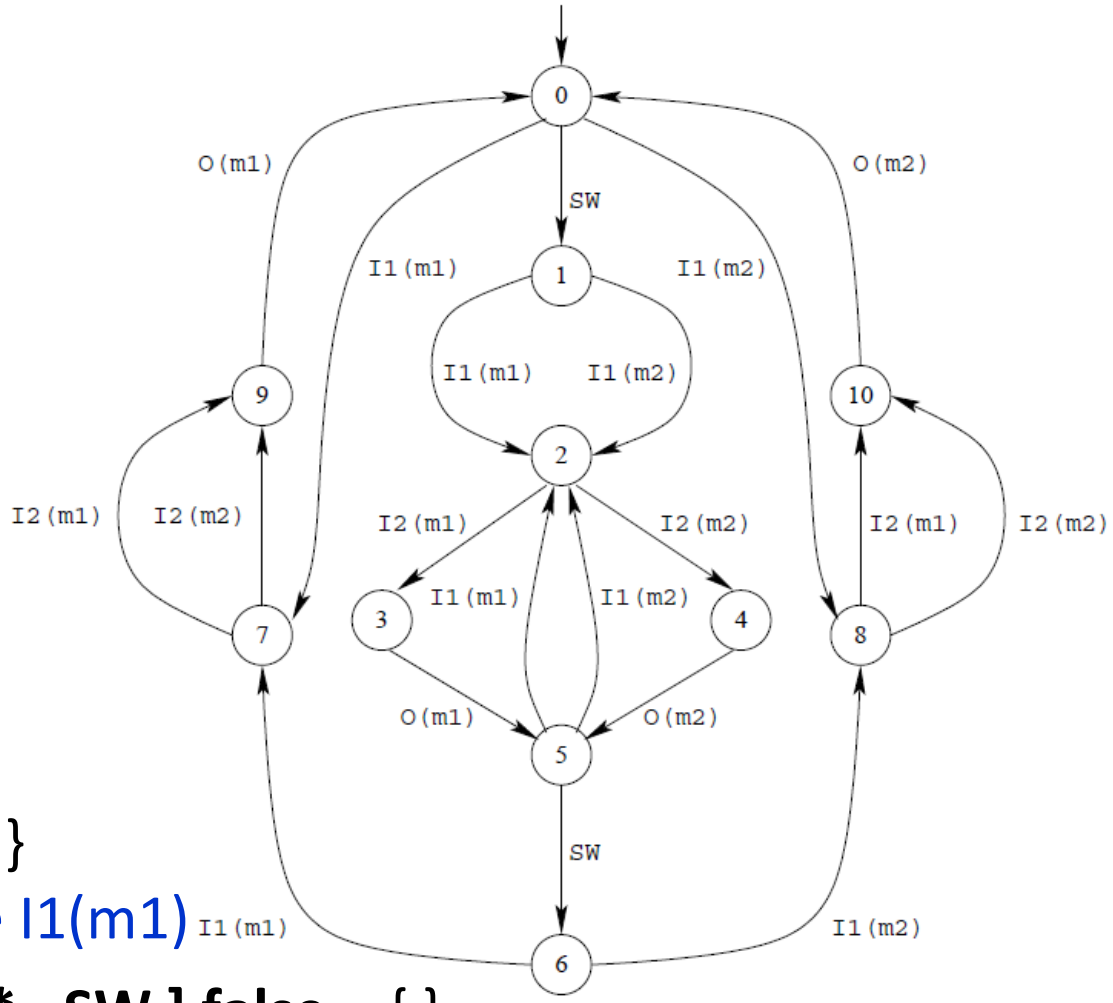
Exam 2018
(exercise – part III)

Radu Mateescu

Inria and LIG / Convecs

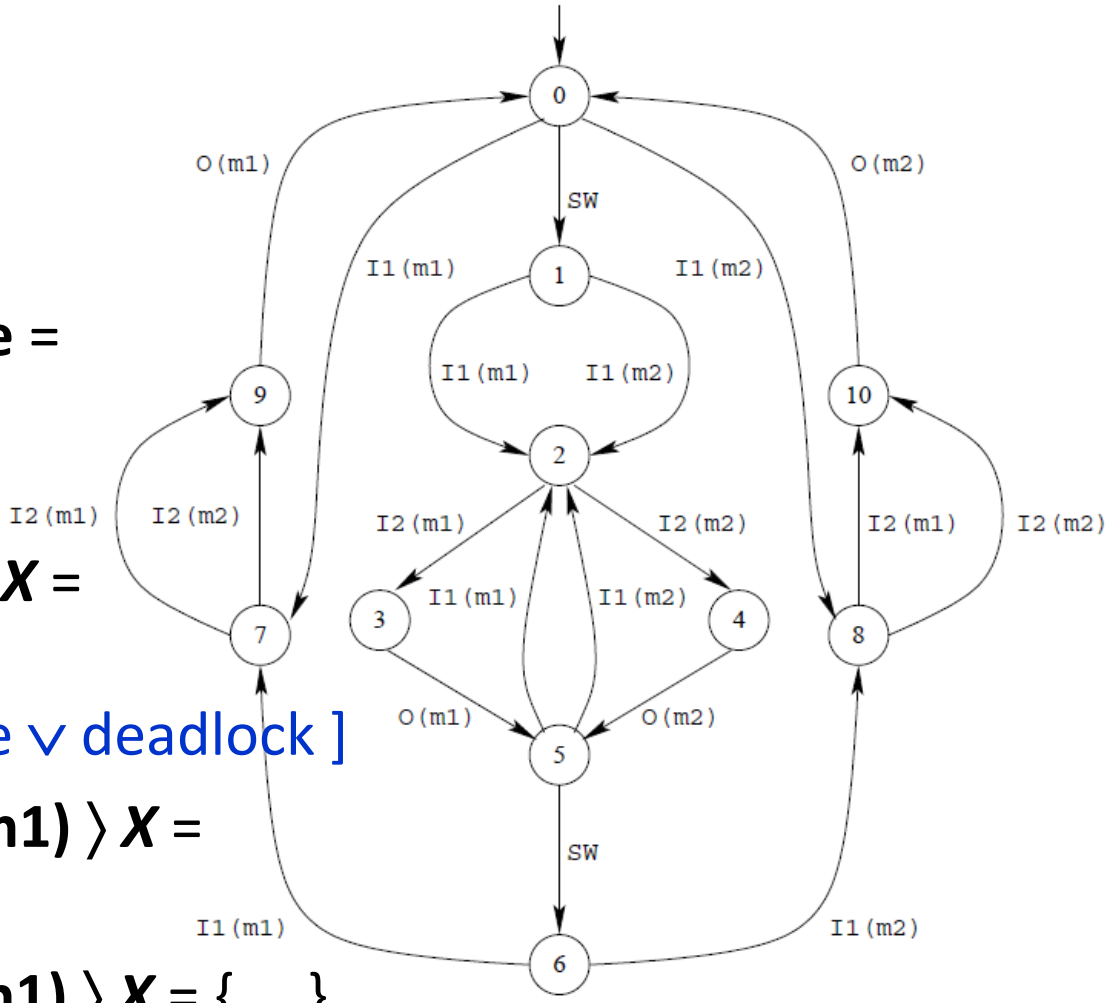
http://convecs.inria.fr

Part III: Temporal logic



1. $\langle$ **O(m1)** $\rangle$ **false** = { }
   // contradiction!
2. $\langle$ **I1(m1)** $\rangle$ **true** = { }
   // can do I1(m1)
3. **[O(m1)] false** = { }
   // cannot do O(m1)
4. **EF** $_{\neg \mathbf{I1(m1)}}$ $\langle$ **O(m1)** $\rangle$ **true** = { }
   // can reach O(m1) before I1(m1)
5. **[SW .** $\neg$ **(O(m1)** $\vee$ **O(m2))* . SW] false** = { }
   // cannot execute SW twice without an O(...) in between

Part III: Temporal logic



6. **[true*] ⟨ true*. SW ⟩ true** =
   { … }
   // always potentially SW
7. **μX . ⟨ SW ⟩ true ∨ [¬SW] X** =
   { … }
   // A [$true_{\neg SW}$ U ⟨ SW ⟩ true ∨ deadlock]
8. **μX . ⟨ I1(m1) . I2(m1) . O(m1) ⟩ X** =
   { } // contradiction!
9. **νX . ⟨ I1(m1) . I2(m1) . O(m1) ⟩ X** = { … }
   // possible infinite repetition of the sequence
5. **νX . ⟨ SW ⟩ X** = { }
   // possible infinite repetition of SW