University of Toronto
CSC2108S – Automated Verification

A Case Study of Simple Mine System With PROMELA/SPIN

APPENDIX

1. Source Code
2. Results of checking properties using SPIN

Su Rong
surong@control.toronto.edu

Yuwei Wang
wangyw@control.toronto.edu

1 Source code of controller of the simple mining system

```
/* Define variables */

bool W_level=0, CH4=0, AF_level=1, CO=0, W_flow=1;
int number=0, C_S=100, preempt=2;
bool generator=0, current_generator=0,next_generator=0;
bool digger, ventilator=0;
bool C_alarm, CO_alarm, AF_alarm, CH4_alarm, WF_alarm;



/* Sensor of water level */

/* When the water level is low, it can change to high randomly.
When  the water level is high, it can only change to low after
the pump generator is started */


proctype Water_level()
{
 do
 :: (W_level == 0) ->

      if
      :: 1 -> W_level = 1
      :: 1 -> W_level = 0
      fi

 :: ( W_level == 1) ->

      if
      :: (generator == 1) -> W_level = 0
      :: else -> skip
      fi
 od
}


/* sensor of CH4 */
```

```
/* When the level of CH4 is 0, it can change to 1 or keep 0
randomly; when the level of CH4 is 1, it will change to 0 after
the ventilator is started */


proctype CH4_level()
{
 do
 :: (CH4 == 0) ->

      if
      :: 1 -> CH4 = 1
      :: 1 -> CH4 = 0
      fi

 :: ( CH4 == 1) ->

      if
      :: (ventilator == 1) -> CH4 = 0
      :: else -> skip
      fi
 od
}


/* Airflow sensor */

/* When the value of airflow sensor is 0, it can change to 1 or
keep 0 randomly; when the value of airflow sensor is 1, it will
change to 0 after the ventilator is started, which means some
air enters */


proctype Airflow()
{
 do
 :: (AF_level == 0) ->

      if
      :: 1 -> AF_level = 1
      :: 1 -> AF_level = 0
      fi

 :: ( AF_level == 1) ->

      if
      :: (ventilator == 1) -> AF_level = 0
      :: else -> skip
      fi
```

```
 od
}



/* CO sensor */

/* When the level of CO is 0, it can change to 1 or keep 0
randomly; when the level of CO is 1, it will change to 0 after
the ventilator is started */


proctype CO_level()
{
 do
 :: (CO == 0) ->

      if
      :: 1 -> CO = 1
      :: 1 -> CO = 0
      fi

 :: ( CO == 1) ->

      if
      :: (ventilator == 1) -> CO = 0
      :: else -> skip
      fi
 od
}

/* Water flow sensor */

proctype Water_flow()
{
 do
 :: W_flow = 1
 :: W_flow = 0
 od
}

/* conveyor sensor */

/* The load remains unchanged or increase by 1 each time */

proctype Conveyor_sensor()
{
```

```
 do
 :: C_S = C_S + 1
 :: C_S = C_S + 0
 od
}

/* The controller for the simple mining system */


proctype Controller()
{
 do

/* when generator is off, it will start to work if the water
level is high or there is user command when the environment is
safe */

:: (generator == 0) ->
       if
       :: ((CH4 == 0) && (preempt == 1 || (W_level == 1 &&
      preempt == 2))) ->generator = 1, current_generator=0,
      next_generator=1;
       :: else -> generator = 0,current_generator=0,
      next_generator=0
       fi;

/*  when  generator  is  on,  it  will  be  shut  down  if  the
environment is not safe or the water level is low or there is
user command when the environment */

      (generator == 1) ->
       if
       :: ((CH4 == 1) || ((CH4 == 0) && ((W_level == 0 &&
      preempt == 2) ||
          (preempt == 0)))) -> generator =0,
      current_generator=1, next_generator=0
       :: else -> generator = 1,current_generator=1,
      next_generator=1
       fi;

/* When the environment is safe, the digger will keep working
*/
    if
    :: ((CH4 == 0) && (W_level == 0) && (C_S < 110)) -> digger
= 1
    :: else -> digger = 0
    fi;
    /* when there is no airflow or the level of CH4 or CO is
high, the ventilator will start to work  */
```

```
    if
    :: ((CH4 == 1) || (AF_level == 0) || (CO == 1)) ->
ventilator = 1
    :: else -> ventilator = 0
    fi;

/* when the level of CH4 is higher the safe level, CH4 alarm
will ring */
    if
    :: (CH4 == 1) -> CH4_alarm = 1
    :: else -> CH4_alarm = 0
    fi ;

/* When there is no air flow, the air flow alarm will ring */

    if
    :: (AF_level == 0) -> AF_alarm = 1
    :: else -> AF_alarm = 0
    fi;

/* When the level of CO is higher than the safe level, the CO
alarm will ring */

    if
    :: (CO == 1) -> CO_alarm = 1
    :: else -> CO_alarm = 0
    fi;

/* When there is no water flow, the water flow alarm will ring
*/
    if
    :: (W_flow == 0) -> WF_alarm = 1
    :: else -> WF_alarm = 0
    fi;

/* Several properties that is implied by the controller */

    ((CH4_alarm == 1)||(AF_alarm == 1)||(CO_alarm == 1)) ->
    assert(ventilator==1);
    (WF_alarm == 1) -> assert(W_flow == 0);

/* If the conveyor alarm rings, the weight of load is more than
110 */

    (C_alarm == 1) -> assert(C_S > 110);
    (digger == 1) -> assert(W_level == 0)
 od
}
```

```
/* High level console */
/* User may send a command when number == 5 */

proctype Console()
{
  do
  ::  (number == 5) ->
                  if
                  :: 1 -> preempt =0; number =0
                  :: 1 -> preempt =0; number =0
                  :: 1 -> preempt =1; number =0
                  fi;
                  number = number +1
  ::  (number < 5) -> preempt =2; number = number +1
  od
}

/* When the weight of the load on the conveyor is more than
110, the conveyor will not work */

proctype Conveyor()
{
 do
 ::  (C_S > 110) -> digger = 0; C_S = 100; C_alarm = 1
 ::  (C_S <= 110) -> C_alarm = 0
 od
}

/* Initialization */

init
{
  run Water_level(); run CH4_level(); run CO_level(); run
Airflow();
  run Water_flow(); run Conveyor_sensor(); run Controller();
  run Console(); run Conveyor()
}
```

Results of checking properties using SPIN

(1) Liveness properties

1. $[]\ ((W_level == 1)\ \text{-> <>}\ (generator == 1))$
2. $[]\ ((W_level == 0)\ \text{-> <>}\ (generator == 0))$
3. $[]\ ((C_S > 110)\ \text{-> <>}\ (C_S <= 110))$
4. $[]\ ((CH4 == 0) \wedge (W_level == 0) \wedge (C_S < 110))$ $\text{--> }(Digger == 1))$

(2) Safety Properties

5. $[]\ (current_generator == 1) \wedge (CH4 == 1) \rightarrow (next_generator == 0))$

6. $[]\ ((current_generator == 0) \wedge (CH4 == 1) \rightarrow (next_generator == 0))$

7. $[]\ ((CH4 == 1) \wedge (W_level == 1) \rightarrow (next_generator == 0))$

8. $[]\ ((CH4 == 0) \wedge (preempt == 1) \wedge (W_level == 0) \rightarrow (next_generator == 1))$
9. $[]\ ((CH4 == 0) \wedge (preempt == 0) \wedge (W_level == 1) \rightarrow (next_generator == 0))$

10. $[]\ ((CH4 == 1) \wedge (preempt == 1) \rightarrow (next_generator == 0))$

11. $[]\ (!\ ((generator == 1) \wedge (digger == 1)))$

2 Results of checking properties using SPIN

- **Liveness properties**

1. [](W_level == 1) -> <> (generator == 1))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1238013, errors: 0
  619007 states, stored
     616 states, matched
  619623 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 828 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
```

```
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

2. [] ((W_level == 0) → <> (generator == 0))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
```

```
      compressed State-vector = 36 byte + 8 byte overhead
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

3. [] (C_S >110) → <> (C_S < 110)

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
```

```
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

- **Safety Properties**

4. [] ((CH4 == 0 ∧ W_level == 0 ∧ C_S < 110) → (digger == 1))

```
eddie% spin -a final_9N

eddie% cc -o run pan.c

eddie% run -a -m □10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)
```

```
Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.194       +memory used for other data structures
268.389     =total actual memory usage

eddie% exit
```

5. [] ((current_generator == 1) ∧ (CH4 == 1) → (next_generator ==0)

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit hints to reduce memory, recompile
with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
```

```
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

6. [] ((current_generator ==0) ∧ (CH4 == 1) →(next_generator == 0))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)
```

```
Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

7. [] ((CH4 == 1) /\ (W_level ==1) → (next_generator == 0))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
```

```
      compressed State-vector = 36 byte + 8 byte overhead
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

8. [] ((CH4 == 0 /\ preempt == 1 /\ W_level ==1) →(next_generator ==1))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
```

```
      compressed State-vector = 36 byte + 8 byte overhead
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

9. [] ((CH4 == 0 ∧ preempt == 0 ∧ W_level ==1) → (next_generator == 0))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
```

```
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

10. [] ((CH4 == 1 /\ preempt ==1) → (next_generator == 0))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
27.238      actual memory usage for states (compression:
68.76%)
```

```
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```

11. [] (!((g == 1) /\ (digger == 1)))

```
eddie% run -a -m10000000

warning: for p.o. reduction to be valid the never claim must be
stutter-closed
(never claims generated from LTL formulae are stutter-closed)
pan: out of memory
      2.68389e+08 bytes used
      102400 bytes more needed
      2.68435e+08 bytes limit (2^MEMCNT)
hints to reduce memory, recompile with:
  -DCOLLAPSE # compression
  -DBITSTATE # supertrace
(Spin Version 2.9.5 -- 18 February 1997)
Warning: Search not completed
      + Partial Order Reduction

Full statespace search for:
      never-claim             +
      assertion violations    + (if within scope of claim)
      acceptance   cycles     + (fairness disabled)
      invalid endstates - (disabled by never-claim)

State-vector 56 byte, depth reached 1237987, errors: 0
  618994 states, stored
     325 states, matched
  619319 transitions (= stored+matched)
       0 atomic steps
hash conflicts: 293 (resolved)
(max size 2^18 states)

Stats on memory usage (in Megabytes):
39.616      equivalent memory usage for states (stored*(State-
vector + overhead))
      compressed State-vector = 36 byte + 8 byte overhead
```

```
27.238      actual memory usage for states (compression:
68.76%)
1.049       +memory used for hash-table (-w18)
240.000     +memory used for DFS stack (-m10000000)
0.193       +memory used for other data structures
268.389     =total actual memory usage
```