

This form is a summary description of the model entitled “Eisenberg-McGuire” proposed for the Model Checking Contest @ Petri Nets. Models can be given in several instances parameterized by scaling parameters. Colored nets can be accompanied by one or many equivalent, unfolded P/T nets. Models are given together with property files (possibly, one per model instance) giving a set of properties to be checked on the model.

Description

This PT net models Eisenberg-McGuire’s algorithm for N processes mutual exclusion problem.

The pseudo code of the algorithm is the following:

```
// variables
constant int N := 4;
shared enum states {IDLE, WAITING, ACTIVE} flag[N - 1] := { IDLE, ..., IDLE};
shared int turn := 0;
int i; // not shared

// code for a process p (p in [0..N-1])
01 loop {
02   repeat {
03     // announce that we need the resource
04     flag[p] := WAITING;
05     // scan processes from the one with the turn up to ourselves.
06     // repeat if necessary until the scan finds all processes idle
07     i := turn;
08     while(i != p) {
09       if(flag[i] != IDLE)
10         i := turn;
11       else
12         i := i + 1 mod N;
13     }
14     // now tentatively claim the resource
15     flag[p] := ACTIVE;
16     // find the first active process besides ourselves, if any
17     i := 0;
18     while (i < n and (i == p or flag[i] != ACTIVE)) {
19       i := i + 1;
20     }
21     // if there were no other active processes, AND if we have the
22     // turn or else whoever has it is idle, then proceed.
23     // Otherwise, repeat the whole sequence.
24   }
25   until (i >= n and (turn == p or flag[turn] == IDLE));
26   // claim the turn and proceed
27   turn := p;
28   // critical section
29   ...
30   // find a process which is not IDLE
31   // (if there are no others, we will find ourselves)
32   i := turn + 1 mod n;
33   while (flag[i] = IDLE) {
34     i := i + 1 mod n;
35   }
36   // give the turn to someone that needs it, or keep it
37   turn := i;
38   // we're finished now
39   flag[p] := IDLE;
40 }
```

Places are named pXX where XX is a line number in the code. Same for transitions. Exceptions are places modeling shared variables and the critical section.

References

Scaling parameter

Parameter name	Parameter description	Chosen parameter values
N	Number of processes competing	3, 4, 5, 6, 7, 8, 9, 10

Size of the model

Parameter	Number of places	Number of transitions	Number of arcs	Number of units	HWB code
$N = 3$	117	216	900	10	2-8-29
$N = 4$	196	448	1952	12	2-10-37
$N = 5$	295	800	3600	13	1-12-50
$N = 6$	414	1296	5976	15	1-14-58
$N = 7$	553	1960	9212	17	1-16-67
$N = 8$	712	2816	13440	19	1-18-75
$N = 9$	891	3888	18792	23	2-21-90
$N = 10$	1090	5200	25400	25	2-23-105

Structural properties

- ordinary** — all arcs have multiplicity one ✓
- simple free choice** — all transitions sharing a common input place have no other input place ✗ (a)
- extended free choice** — all transitions sharing a common input place have the same input places ✗ (b)
- state machine** — every transition has exactly one input place and exactly one output place ✗ (c)
- marked graph** — every place has exactly one input transition and exactly one output transition ✗ (d)
- connected** — there is an undirected path between every two nodes (places or transitions) ✓ (e)
- strongly connected** — there is a directed path between every two nodes (places or transitions) ✓ (f)
- source place(s)** — one or more places have no input transitions ✗ (g)
- sink place(s)** — one or more places have no output transitions ✗ (h)
- source transition(s)** — one or more transitions have no input places ✗ (i)
- sink transitions(s)** — one or more transitions have no output places ✗ (j)
- loop-free** — no transition has an input place that is also an output place ✗ (k)
- conservative** — for each transition, the number of input arcs equals the number of output arcs ✓ (l)
- subconservative** — for each transition, the number of input arcs equals or exceeds the number of output arcs ✓ (m)
- nested units** — places are structured into hierarchically nested sequential units⁽ⁿ⁾ ✓

Behavioural properties

- safe** — in every reachable marking, there is no more than one token on a place ✓ (o)
- dead place(s)** — one or more places have no token in any reachable marking ✓ (p)
- dead transition(s)** — one or more transitions cannot fire from any reachable marking ✓ (q)
- deadlock** — there exists a reachable marking from which no transition can be fired ✗ (r)
- reversible** — from every reachable marking, there is a transition path going back to the initial marking ?

(a) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(b) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(c) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(d) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(e) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(f) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(g) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(h) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(i) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(j) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(k) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(l) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(m) stated by [CÆSAR.BDD](#) version 3.7 on all 8 instances (3, 4, 5, 6, 7, 8, 9, 10, and 11).

(n) the definition of Nested-Unit Petri Nets (NUPN) is available from <http://mcc.lip6.fr/nupn.php>

(o) safe by construction – stated by [PNML2NUPN](#) 3.2.0.

(p) stated by [CÆSAR.BDD](#) version 3.7 to be false on 2 instance(s) out of 8, and unknown on the remaining 6 instance(s).

(q) stated by [CÆSAR.BDD](#) version 3.7 to be false on 2 instance(s) out of 8, and unknown on the remaining 6 instance(s).

(r) stated by [CÆSAR.BDD](#) version 3.7 to be false on 2 instance(s) out of 8, and unknown on the remaining 6 instance(s).

live — for every transition t , from every reachable marking, one can reach a marking in which t can fire

Size of the marking graphs

Parameter	Number of reachable markings	Number of transition firings	Max. number of tokens per place	Max. number of tokens per marking
$N = 3$	31,265 ^(s)	93,795	1	7 ^(t)
$N = 4$	1,762,378 ^(u)	7,049,512	1	9 ^(v)
$N = 5$	114,106,399	570,531,995	1 ^(w)	11 ^(x)
$N = 6$	$\geq 7.46519e+08$ ^(y)	?	1 ^(z)	13 ^(aa)
$N = 7$	$\geq 1.4107e+09$ ^(ab)	?	1 ^(ac)	15 ^(ad)
$N = 8$	$\geq 1.64208e+10$ ^(ae)	?	1 ^(af)	17 ^(ag)
$N = 9$	$\geq 6.6762e+10$ ^(ah)	?	1 ^(ai)	19 ^(aj)
$N = 10$	$\geq 4.96501e+10$ ^(ak)	?	1 ^(al)	21 ^(am)

Other properties

- State property: at each reachable marking, there is at most process in critical section.

$$\sum_{i \in \{0..N-1\}} critical_section_i \leq 1$$

- LTL property: there infinitely often is a process in critical section.

$$\ll (\langle \langle \sum_{i \in \{0..N-1\}} critical_section_i == 1 \rangle \rangle)$$

(s) stated by [CÆSAR.BDD](#) version 3.7.
 (t) number of initial tokens, because the net is conservative.
 (u) stated by [CÆSAR.BDD](#) version 3.7.
 (v) number of initial tokens, because the net is conservative.
 (w) stated by [PNML2NUPN](#) 3.2.0.
 (x) number of initial tokens, because the net is conservative.
 (y) stated by [CÆSAR.BDD](#) version 3.7.
 (z) stated by [PNML2NUPN](#) 3.2.0.
 (aa) number of initial tokens, because the net is conservative.
 (ab) stated by [CÆSAR.BDD](#) version 3.7.
 (ac) stated by [PNML2NUPN](#) 3.2.0.
 (ad) number of initial tokens, because the net is conservative.
 (ae) stated by [CÆSAR.BDD](#) version 3.7.
 (af) stated by [PNML2NUPN](#) 3.2.0.
 (ag) number of initial tokens, because the net is conservative.
 (ah) stated by [CÆSAR.BDD](#) version 3.7.
 (ai) stated by [PNML2NUPN](#) 3.2.0.
 (aj) number of initial tokens, because the net is conservative.
 (ak) stated by [CÆSAR.BDD](#) version 3.7.
 (al) stated by [PNML2NUPN](#) 3.2.0.
 (am) number of initial tokens, because the net is conservative.