

This form is a summary description of the model entitled “FlexibleBarrier” 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 model describes a barrier algorithm that synchronizes several concurrent processes. The number of concurrent processes evolves dynamically over time. Between two successive synchronizations, processes can be killed or can fork new processes: all live processes have to be synchronized by the barrier. This algorithm is used in cooperative kernels, a novel technique that allows safe multitasking for irregular data-parallel algorithms on GPUs.

This collection of P/T nets was derived from an LNT model of the Flexible Barrier. Each instance was first translated to LOTOS, and then to an interpreted Petri net using the [CADP](#) toolbox. Finally, a P/T net was obtained by stripping out all data-related information (variables, types, assignments, guards, etc.) from the interpreted Petri net, leading to a NUPN (Nested-Unit Petri Net) model translated to PNML using the [CÆSAR.BDD](#) tool.

Each instance of the model is parameterized by the maximal number N of concurrent processes that synchronize on the barrier. Each instance is also parameterized by its version V , which specifies how the NUPN has been produced from the LOTOS specification. V is either equal to “ a ” if the NUPN has been generated *after* applying all the structural and data-flow optimizations of the [CÆSAR](#) compiler for LOTOS, or to “ b ” if the NUPN has been generated *before* these optimizations.

References

Scaling parameter

Parameter name	Parameter description	Chosen parameter values
(N, V)	N is the maximal number of concurrent processes and V is the version defined above	$\{4, 6, 8, 10, 12, 14, 16, 18, 20, 22\} \times \{a, b\}$

Size of the model

Parameter	Number of places	Number of transitions	Number of arcs	Number of units	HWB code
$N = 04, V = a$	51	88	309	7	2-6-19
$N = 04, V = b$	268	305	743	11	5-6-39
$N = 06, V = a$	75	154	599	9	2-8-27
$N = 06, V = b$	542	621	1533	15	7-8-60
$N = 08, V = a$	99	236	985	11	2-10-35
$N = 08, V = b$	920	1057	2627	19	9-10-76
$N = 10, V = a$	123	334	1467	13	2-12-43
$N = 10, V = b$	1402	1613	4025	23	11-12-94
$N = 12, V = a$	147	448	2045	15	2-14-51
$N = 12, V = b$	1988	2289	5727	27	13-14-111
$N = 14, V = a$	171	578	2719	17	2-16-59
$N = 14, V = b$	2678	3085	7733	31	15-16-140
$N = 16, V = a$	195	724	3489	19	2-18-67
$N = 16, V = b$	3472	4001	10043	35	17-18-159
$N = 18, V = a$	219	886	4355	21	2-20-75
$N = 18, V = b$	4370	5037	12657	39	19-20-177
$N = 20, V = a$	243	1064	5317	23	2-22-83
$N = 20, V = b$	5372	6193	15575	43	21-22-195
$N = 22, V = a$	267	1258	6375	25	2-24-91
$N = 22, V = b$	6478	7469	18797	47	23-24-214

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⁾	✓

(a) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(b) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(c) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(d) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(e) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(f) from place 1 one cannot reach place 0.

(g) place 0 is a source place.

(h) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(i) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(j) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(k) stated by [CÆSAR.BDD](#) version 2.7 to be true on 10 instance(s) out of 20, and false on the remaining 10 instance(s).

(l) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

(m) stated by [CÆSAR.BDD](#) version 2.7 on all 20 instances (10 values of $N \times 2$ values of V).

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

Behavioural properties

- safe** — *in every reachable marking, there is no more than one token on a place* ✓ ^(o)
deadlock — *there exists a reachable marking from which no transition can be fired* ? ^(p)
reversible — *from every reachable marking, there is a transition path going back to the initial marking* ?
quasi-live — *for every transition t , there exists a reachable marking in which t can fire* ? ^(q)
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
$N = 04, V = a$	20737 ^(r)	?	1	6
$N = 04, V = b$	$\geq 3.66789e+08$ ^(s)	?	1 ^(t)	$\in [2, 6]$ ^(u)
$N = 06, V = a$	$2.98598e+06$ ^(v)	?	1	8
$N = 06, V = b$	$\geq 4.63158e+10$ ^(w)	?	1 ^(x)	$\in [2, 8]$ ^(y)
$N = 08, V = a$	$4.29982e+08$ ^(z)	?	1	10
$N = 08, V = b$	$\geq 2.87705e+13$ ^(aa)	?	1 ^(ab)	$\in [2, 10]$ ^(ac)
$N = 10, V = a$	$6.19174e+10$ ^(ad)	?	1	12
$N = 10, V = b$	$\geq 1.14075e+17$ ^(ae)	?	1 ^(af)	$\in [2, 12]$ ^(ag)
$N = 12, V = a$	$8.9161e+12$ ^(ah)	?	1	14
$N = 12, V = b$	$\geq 6.12466e+20$ ^(ai)	?	1 ^(aj)	$\in [2, 14]$ ^(ak)
$N = 14, V = a$	$1.28392e+15$ ^(al)	?	1	16
$N = 14, V = b$	$\geq 3.04107e+15$ ^(am)	?	1 ^(an)	$\in [2, 16]$ ^(ao)
$N = 16, V = a$	$1.84884e+17$ ^(ap)	?	1	18
$N = 16, V = b$	$\geq 2.19392e+17$ ^(aq)	?	1 ^(ar)	$\in [2, 18]$ ^(as)
$N = 18, V = a$	$2.66233e+19$ ^(at)	?	1	20
$N = 18, V = b$	$\geq 1.51447e+19$ ^(au)	?	1 ^(av)	$\in [2, 20]$ ^(aw)
$N = 20, V = a$	$3.83376e+21$ ^(ax)	?	1	22
$N = 20, V = b$	$\geq 1.00937e+21$ ^(ay)	?	1 ^(az)	$\in [2, 22]$ ^(ba)
$N = 22, V = a$	$5.52061e+23$ ^(bb)	?	1	24
$N = 22, V = b$	$\geq 6.53756e+22$ ^(bc)	?	1 ^(bd)	$\in [2, 24]$ ^(be)

- ^(o) safe by construction – stated by the [CÆSAR](#) compiler.
^(p) stated by [CÆSAR.BDD](#) version 2.7 to be false on 10 instance(s) out of 20, and unknown on the remaining 10 instance(s).
^(q) stated by [CÆSAR.BDD](#) version 2.7 to be true on 11 instance(s) out of 20, and unknown on the remaining 9 instance(s).
^(r) stated by [CÆSAR.BDD](#) version 2.7.
^(s) stated by [CÆSAR.BDD](#) version 2.7.
^(t) stated by the [CÆSAR](#) compiler.
^(u) lower and upper bounds given by the number of initial tokens and the number of leaf units.
^(v) stated by [CÆSAR.BDD](#) version 2.7.
^(w) stated by [CÆSAR.BDD](#) version 2.7.
^(x) stated by the [CÆSAR](#) compiler.
^(y) lower and upper bounds given by the number of initial tokens and the number of leaf units.
^(z) stated by [CÆSAR.BDD](#) version 2.7.
^(aa) stated by [CÆSAR.BDD](#) version 2.7.
^(ab) stated by the [CÆSAR](#) compiler.
^(ac) lower and upper bounds given by the number of initial tokens and the number of leaf units.
^(ad) stated by [CÆSAR.BDD](#) version 2.7.
^(ae) stated by [CÆSAR.BDD](#) version 2.7.
^(af) stated by the [CÆSAR](#) compiler.
^(ag) lower and upper bounds given by the number of initial tokens and the number of leaf units.
^(ah) stated by [CÆSAR.BDD](#) version 2.7.
^(ai) stated by [CÆSAR.BDD](#) version 2.7.
^(aj) stated by the [CÆSAR](#) compiler.
^(ak) lower and upper bounds given by the number of initial tokens and the number of leaf units.
^(al) stated by [CÆSAR.BDD](#) version 2.7.
^(am) stated by [CÆSAR.BDD](#) version 2.7.
^(an) stated by the [CÆSAR](#) compiler.

-
- (ao) lower and upper bounds given by the number of initial tokens and the number of leaf units.
 - (ap) stated by [CÆSAR.BDD](#) version 2.7.
 - (aq) stated by [CÆSAR.BDD](#) version 2.7.
 - (ar) stated by the [CÆSAR](#) compiler.
 - (as) lower and upper bounds given by the number of initial tokens and the number of leaf units.
 - (at) stated by [CÆSAR.BDD](#) version 2.7.
 - (au) stated by [CÆSAR.BDD](#) version 2.7.
 - (av) stated by the [CÆSAR](#) compiler.
 - (aw) lower and upper bounds given by the number of initial tokens and the number of leaf units.
 - (ax) stated by [CÆSAR.BDD](#) version 2.7.
 - (ay) stated by [CÆSAR.BDD](#) version 2.7.
 - (az) stated by the [CÆSAR](#) compiler.
 - (ba) lower and upper bounds given by the number of initial tokens and the number of leaf units.
 - (bb) stated by [CÆSAR.BDD](#) version 2.7.
 - (bc) stated by [CÆSAR.BDD](#) version 2.7.
 - (bd) stated by the [CÆSAR](#) compiler.
 - (be) lower and upper bounds given by the number of initial tokens and the number of leaf units.