Fabrice Kordon
Fabrice.Kordon@lip6.fr

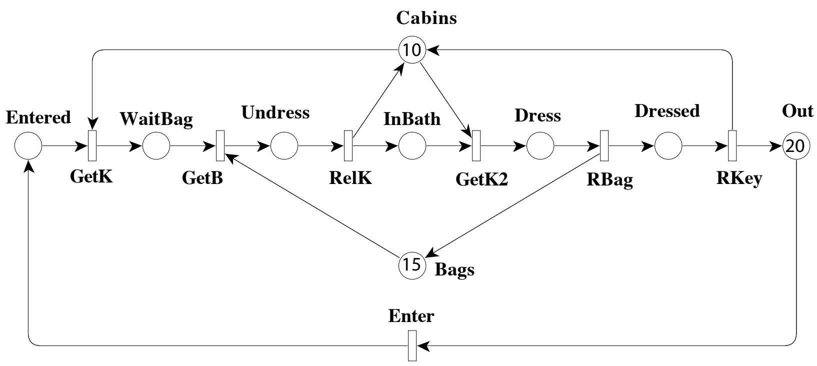
This form is a summary description of the model entitled "SwimmingPool" proposed for the Model Checking Contest @ Petri Nets. Models can be given in several instances parameterized by scaling parameters. Colored nets can be accompanied by one or many equivalent, unfolded P/T nets. Models are given together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

Description

In this classical example, the director of a swimming pool has established a protocol to use the pool. The protocol is described as follows:

- S_1 A user gets into the building and gets a key for a cabin,
- S_2 He then ask for a bag to put his clothes on and then uses the cabin to undress and get his swimming suit,
- S_3 He then return the key and can enjoy the swimming pool,
- S_4 He gets out the swimming pool and ask for the key of a new cabin,
- S_5 He dresses again, and then give back his bag,
- S_6 He gives back the key of the cabin and then leaves the building.

The system has a scaling parameter N from which the numbers of cabins, bags, and persons in the swimming pool are deduced. For a given value N, we consider $N \times 10$ cabins, $N \times 15$ bags and $N \times 20$ persons.



Graphical representation for $N = 1$

Scaling parameter

Parameter name	Parameter description	Chosen parameter values
N	N, a parameter from which the numbers of cabins, bags, and persons in the pool are deduced. [(a)].	$N = 1$, $N = 2$, $N = 3$, $N = 4$, $N = 5$, $N = 6$, $N = 7$, $N = 8$, $N = 9$, $N = 10$

Size of the model

Although the model is parameterized, its size does not depend on parameter values.

number of places:	9
number of transitions:	7
number of arcs:	20

[(a)] These parameters affect the initial marking and thus do not impact the size of the model.

Fabrice Kordon
Fabrice.Kordon@lip6.fr

Structural properties

ordinary — *all arcs have multiplicity one* ✔
simple free choice — *all (different) transitions with a shared input place have no other input place* ✗ [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]
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* ✔ [q]
quasi-live — *for every transition t, there exists a reachable marking in which t can fire* ?
live — *for every transition t, from every reachable marking, one can reach a marking in which t can fire* ?

---

[b] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[c] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[d] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[e] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[f] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[g] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[h] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[i] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[j] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[k] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[l] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[m] stated by CÆSAR.BDD version 2.2 on all 10 instances (1, 2, 3, 4, 5, 6, 7, 8, 9, 10).
[n] the definition of Nested-Unit Petri Nets (NUPN) is available from `http://mcc.lip6.fr/nupn.php`
[o] By construction of the model (The initial marking is not safe)..
[p] If there are more bags than cabins only..
[q] If there are more bags than cabins only..

Fabrice Kordon
Fabrice.Kordon@lip6.fr

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
$N = 1$	89 621 (r)	450 003 (s)	?	≥ 45 (t)
$N = 2$	3 408 031 (u)	19 929 811 (v)	?	≥ 90 (w)
$N = 3$	?	?	?	≥ 135 (x)
$N = 4$	?	?	?	≥ 180 (y)
$N = 5$	?	?	?	≥ 225 (z)
$N = 6$	?	?	?	≥ 270 (aa)
$N = 7$	?	?	?	≥ 315 (ab)
$N = 8$	?	?	?	≥ 360 (ac)
$N = 9$	?	?	?	≥ 405 (ad)
$N = 10$	?	?	?	≥ 450 (ae)

Other properties

If the number of bags is greater than the number of cabins, this model does not exhibit any deadlock. Otherwise, there is a deadlock.

(r) computed by PROD in December 2014.
(s) computed by PROD in December 2014.
(t) lower bound given by the number of initial tokens.
(u) computed by PROD in December 2014.
(v) computed by PROD in December 2014.
(w) lower bound given by the number of initial tokens.
(x) lower bound given by the number of initial tokens.
(y) lower bound given by the number of initial tokens.
(z) lower bound given by the number of initial tokens.
(aa) lower bound given by the number of initial tokens.
(ab) lower bound given by the number of initial tokens.
(ac) lower bound given by the number of initial tokens.
(ad) lower bound given by the number of initial tokens.
(ae) lower bound given by the number of initial tokens.