Benoît Barbot and Marta Kwiatkowska
barbot@u-pec.fr

This form is a summary description of the model entitled "DNAwalker" 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

DNAwalker stands for "*DNA Walker Circuits*".

Molecular programming is an emerging field concerned with building synthetic biomolecular computing devices at the nanoscale, for example from DNA or RNA molecules. Many promising applications have been proposed, ranging from diagnostic biosensors and nanorobots to synthetic biology, but prohibitive complexity and imprecision of experimental observations makes reliability of molecular programs difficult to achieve. This paper advocates the development of design automation methodologies for molecular programming, highlighting the role of quantitative verification in this context. We focus on DNA 'walker' circuits, in which molecules can be programmed to traverse tracks placed on a DNA origami tile, taking appropriate decisions at junctions and reporting the outcome when reaching the end of the track.

Here we present a modelling of DNA walker as Generalised Stochastic Petri Net (GSPN) the full description of the modelling can be found in [1].

In [2], DNA walkers were modelled in the native language of PRISM, which represents each walker circuit as a synchronised parallel composition of reactive modules, each specified using guarded commands whose updates are annotated with rates. Since DNA walker circuits are planar, generalised stochastic Petri (GSPN) nets are well suited to their modelling, with the layout of the GSPN closely corresponding to the layout of the original circuit: the state of each anchorage is modelled using an independent place, while the steps of the walker are modelled with transitions that are exponentially distributed.

The blocking mechanism used to steer the movement of the walker may fail with some probability; this failure occurs before the walker is released and thus before any walker movement. In PRISM, this is modelled with transitions with very high rates (a billion times larger than walker movement rates), which makes the computation intractable when uniformisation is used. In GSPN this blocking mechanism is modelled using instantaneous transitions.

More precisely, each DNA walker circuit comprises several tracks (sequences of anchorages) and transitions correspond to walker taking a step from one anchorage to another nearby. The states of each anchorage are modelled as follows:

- Each anchorage is modelled with a single place, to preserve the layout of the original circuit placement. The relative placement of each place corresponds to that of the corresponding anchorage on the origami.
- Intact anchorages are modelled with places containing one token.
- Anchorages where the top has melted away are modelled by empty places.
- The anchorage to which the walker is attached is modelled by a place with two tokens.
- Blocked anchorages are modelled like anchorages where the top has melted away, with empty places.

Fig. 1 illustrates a transition encoding a displacement reaction between two anchorages a and b. Place a encodes the anchorage to which to walker is currently bound. Place b encodes an intact, unblocked anchorage. The transition consumes two tokens in the place corresponding to a and one token in the place corresponding to b, and produces two tokens in the place corresponding to b. Indeed, after the transition is fired the place corresponding to a is left empty, which models the anchorage where the top has melted away.

The walker may move between two anchorages that are sufficiently close. Each such movement is modelled with independent transitions. The rate of each transition depends on the distance between the two anchorages.

Blocked anchorages do not initially contain tokens. In order to model the possibility of failure of the blocking mechanism, a place with initially one token and two immediate concurrent transitions are added to each blocked anchorage. Fig. 2 illustrates this. With failure probability $f = 0.3$, a token is added to the place for anchorage a. In this case the anchorage is no longer blocked.

Our modelling approach ensures modularity of the design and that the layout of the Petri net closely resembles that of the original walker system. Different circuits can be composed together easily by merging together initial and final anchorages

Benoît Barbot and Marta Kwiatkowska
barbot@u-pec.fr

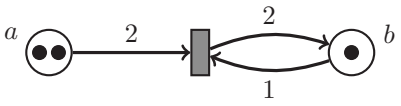


Figure 1: Transition modelling movement of the walker from anchorage a to anchorage b.

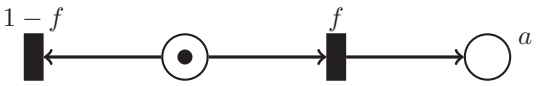
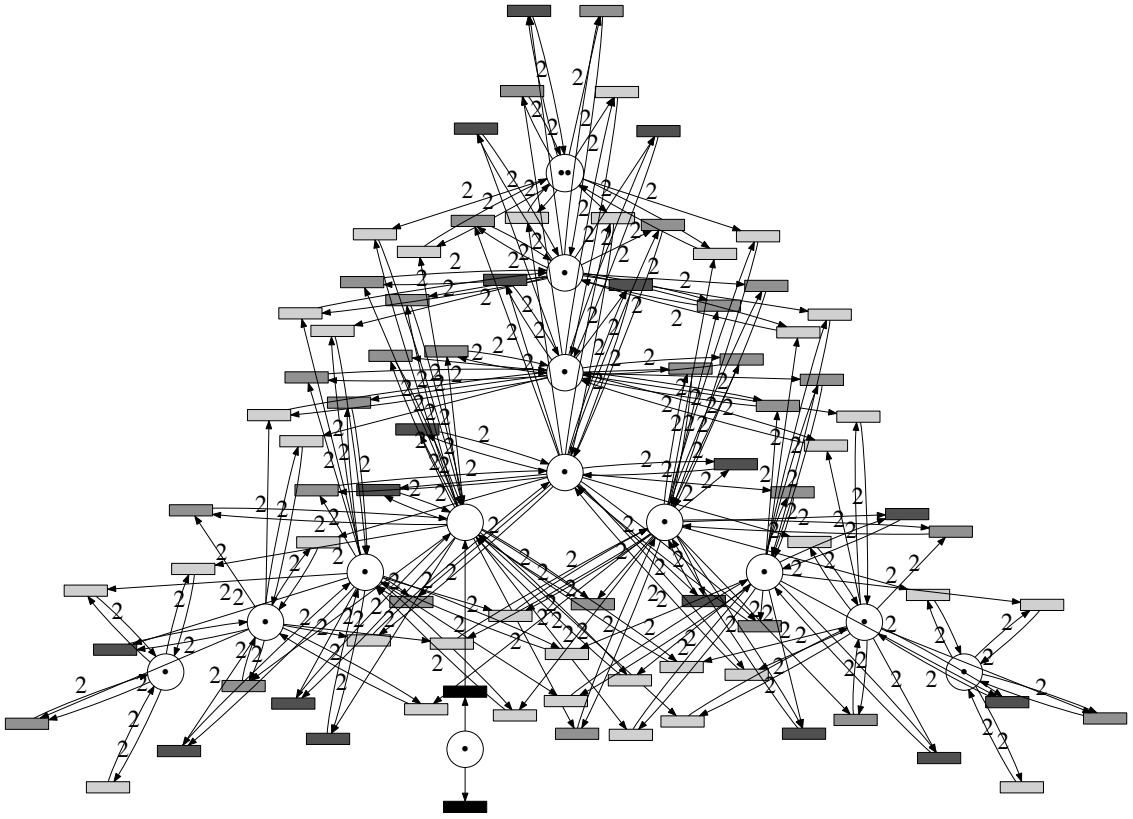


Figure 2: Two transition model of the failure of the blocking mechanism of anchorage a.

and by adding transitions between places encoding nearby transitions. Additional behaviours of the circuit, such as a missing anchorage, may be added easily by adding or removing tokens.



Graphical representation for the topology $N = 1$

References

[1] B. Barbot and M. Kwiatkowska. On Quantitative Modelling and Verification of DNA Walker Circuits Using Stochastic Petri Nets. In *Proc. 36th International Conference on Application and Theory of Petri Nets and Concurrency*, volume 9115 of Lecture Notes in Computer Science, pages 1–32, Springer International Publishing. 2015.

[2] F. Dannenberg, M. Kwiatkowska, C. Thachuk, and A. Turberfield. DNA walker circuits: computational potential, design, and verification. In D. Soloveichik and B. Yurke, editors, *Proc. 19th International Conference on DNA Computing and Molecular Programming (DNA 19)*, volume 8141 of LNCS, pages 31–45. Springer, 2013.

Benoît Barbot and Marta Kwiatkowska
barbot@u-pec.fr

Scaling parameter

Parameter name	Parameter description	Chosen parameter values
N	The topology of the DNA track on which the walker evolves.	$N = 1 \rightarrow$ track12Block1 $N = 2 \rightarrow$ track12Block2 $N = 3 \rightarrow$ track12BlockBoth $N = 4 \rightarrow$ track28LL $N = 5 \rightarrow$ track28LR $N = 8 \rightarrow$ ringLL $N = 9 \rightarrow$ ringLR $N = 12 \rightarrow$ ringLLLarge $N = 14 \rightarrow$ ringLRLarge $N = 16 \rightarrow$ redundantChoice01 $N = 18 \rightarrow$ losangeBlock

Size of the model

Parameter	Number of places	Number of transitions	Number of arcs
$N = 1$	13	82	241
$N = 2$	14	84	244
$N = 3$	14	84	244
$N = 4$	34	250	728
$N = 5$	34	250	728
$N = 8$	27	260	760
$N = 9$	27	260	760
$N = 12$	33	312	916
$N = 14$	33	312	916
$N = 16$	43	490	1438
$N = 18$	164	3697	10898

Structural properties

ordinary — *all arcs have multiplicity one* ..... no
simple free choice — *all transitions sharing a common input place have no other input place* ..... no [a]
extended free choice — *all transitions sharing a common input place have the same input places* ..... no [b]
state machine — *every transition has exactly one input place and exactly one output place* ..... no [c]
marked graph — *every place has exactly one input transition and exactly one output transition* ..... no [d]
connected — *there is an undirected path between every two nodes (places or transitions)* ..... yes [e]
strongly connected — *there is a directed path between every two nodes (places or transitions)* ..... no [f]
source place(s) — *one or more places have no input transitions* ..... ? [g]
sink place(s) — *one or more places have no output transitions* ..... no [h]
source transition(s) — *one or more transitions have no input places* ..... no [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* ..... no [k]
conservative — *for each transition, the number of input arcs equals the number of output arcs* ..... no [l]

[a] the net is not ordinary.
[b] the net is not ordinary.
[c] the net is not ordinary.
[d] the net is not ordinary.
[e] stated by CÆSAR.BDD version 2.6 on all 11 instances (see all aforementioned topology values).
[f] stated by CÆSAR.BDD version 2.6 on all 11 instances (see all aforementioned topology values).
[g] stated by CÆSAR.BDD version 2.6 to be true on all 11 instances (see all aforementioned topology values).
[h] stated by CÆSAR.BDD version 2.6 on all 11 instances (see all aforementioned topology values).
[i] stated by CÆSAR.BDD version 2.6 on all 11 instances (see all aforementioned topology values).
[j] stated by CÆSAR.BDD version 2.6 to be true on all 11 instance(s) (see all aforementioned topology values).
[k] stated by CÆSAR.BDD version 2.6 on all 11 instances (see all aforementioned topology values).
[l] stated by PNML2NUPN 1.5.3 on all 11 instances (see all aforementioned topology values).

Benoît Barbot and Marta Kwiatkowska
barbot@u-pec.fr

subconservative — *for each transition, the number of input arcs equals or exceeds the number of output arcs* ..... yes [m]
nested units — *places are structured into hierarchically nested sequential units* [n] ..... no

Behavioural properties

safe — *in every reachable marking, there is no more than one token on a place* ..... no [o]
dead place(s) — *one or more places have no token in any reachable marking* ..... ?
dead transition(s) — *one or more transitions cannot fire from any reachable marking* ..... ?
deadlock — *there exists a reachable marking from which no transition can be fired* ..... yes
reversible — *from every reachable marking, there is a transition path going back to the initial marking* ..... no
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
$N = 1$	3795	?	2	13 [p]
$N = 2$	5459	?	2	13 [q]
$N = 3$	5248	?	2	13 [r]
$N = 4$	432,884,827	?	2	29 [s]
$N = 5$	435,340,831	?	2	29 [t]
$N = 8$	27,950,678	?	2	22 [u]
$N = 9$	28,209,796	?	2	22 [v]
$N = 12$	1,885,372,776	?	2	28 [w]
$N = 14$	1,860,879,029	?	2	28 [x]
$N = 16$	?	?	2	34 [y]
$N = 18$	?	?	2	101 [z]

[m] stated by PNML2NUPN 1.5.3 on all 11 instances (see all aforementioned topology values).
[n] the definition of Nested-Unit Petri Nets (NUPN) is available from http://mcc.lip6.fr/nupn.php
[o] in the initial marking, there exists one place containing 2 tokens.
[p] number of initial tokens, because the net is sub-conservative.
[q] number of initial tokens, because the net is sub-conservative.
[r] number of initial tokens, because the net is sub-conservative.
[s] number of initial tokens, because the net is sub-conservative.
[t] number of initial tokens, because the net is sub-conservative.
[u] number of initial tokens, because the net is sub-conservative.
[v] number of initial tokens, because the net is sub-conservative.
[w] number of initial tokens, because the net is sub-conservative.
[x] number of initial tokens, because the net is sub-conservative.
[y] number of initial tokens, because the net is sub-conservative.
[z] number of initial tokens, because the net is sub-conservative.