Silvano Dal Zilio
dalzilio@laas.fr

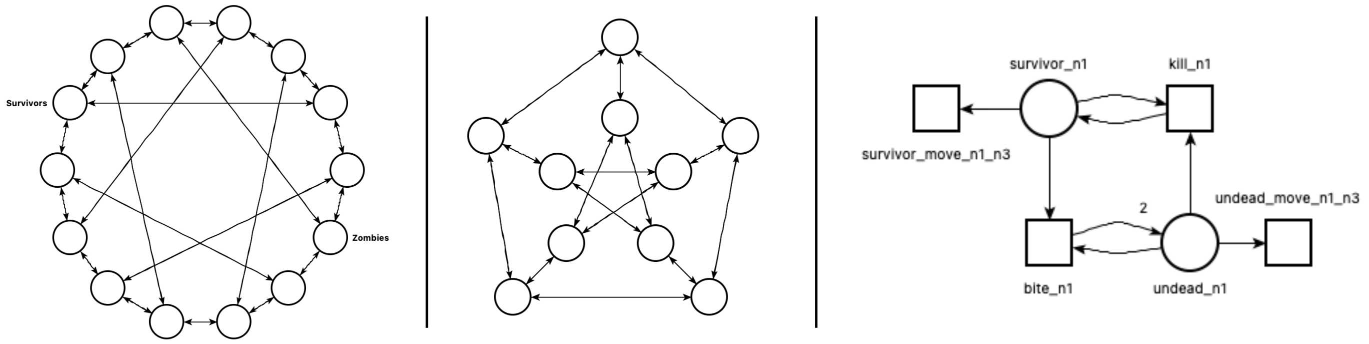
This form is a summary description of the model entitled "ZombiesAndSurvivors" proposed for the Model Checking Contest @ Petri Nets. Models can be given in several instances parameterized by scaling parameters. Colored nets can be accompanied by one or many equivalent, unfolded P/T nets. Models are given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

Description

Zombies and Survivors [1] is a typical example of pursuit-evasion game played on graph structures, such as Cop and Robbers. These types of games are used in graph theory to define strategies, different notions of optimal play, and as a tool for studying computational complexity.

The game is played on a fully connected, undirected graph where nodes correspond to positions. In our version of this game, human survivors and zombies (or undead) can freely move around the graph, and the humans can defend themselves. When a zombie and a survivor are on the same node, a zombie can either bite a human, which then transforms into a zombie, or the human can eliminate the undead.

Our model uses two places for each node in the graph—one for undead and the other for humans—and different transitions for the bite, kill and move events. We consider three main categories of graphs, parametrized by their length / number of nodes (L): `Circular` is a simple cycle of size L, whereas `Heawood` and `Kneser` are families of graphs with small degree but large girth. The nodes in the Kneser graph (N, K) corresponds to the subsets, of size K, of a set of N elements. So they have $\binom{N}{K}$ nodes. The size of the "Heawood"-like graph is $3\,N - 1$, for some $N \geq 5$. The final two parameters are the initial number of survivors (X) and zombies (Y), even though we chose to have $X = Y$ in all our instances. Both groups start on opposite sides of the graph.



Left: the Heawood graph (for $L = 14$); Middle: the Kneser graph (for $N = 5$ and $K = 2$); Right: the Petri net associated with node `n1` and an edge from `n1` to `n3`. The number of "move" transitions depends on the degree of the node.

References

1. Fitzpatrick, Shannon, et al. *"A deterministic version of the game of zombies and survivors on graphs."* Discrete Applied Mathematics 213 (2016): 1-12.

Scaling parameter

Parameter name	Parameter description	Chosen parameter values
Circular (L, X, Y) or Heawood (L, X, Y) or Kneser (L, N, K, X, Y)	L is the number of nodes in the underlying graph, whereas X and Y are the initial numbers of survivors and undead. For Kneser graphs, the value of L is $\binom{N}{K}$.	Circular $(4, 5, 5)$, $(4, 100, 100)$, $(8, 100, 100)$, $(32, 50, 50)$, $(64, 10, 10)$ Heawood $(14, 5, 5)$, $(14, 60, 60)$, $(17, 25, 25)$, $(17, 50, 50)$, $(38, 15, 15)$ Kneser $(21, 7, 2, 5, 5)$, $(21, 7, 2, 60, 60)$, $(21, 7, 2, 80, 80)$, $(35, 7, 3, 60, 60)$

Silvano Dal Zilio
dalzilio@laas.fr

Size of the model

Parameter	Number of places	Number of transitions	Number of arcs
Circular (L, X, Y)	$2\,L$	$6\,L$	$14\,L$
Heawood (L, X, Y)	$2\,L$	$8\,L$	$18\,L$
Kneser (L, N, K, X, Y)	$2\,L$	?	?

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)* .... yes [(f)]
source place(s) — *one or more places have no input transitions* .... no [(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* .... no [(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)]
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* .... no [(p)]
dead transition(s) — *one or more transitions cannot fire from any reachable marking* .... no [(q)]
deadlock — *there exists a reachable marking from which no transition can be fired* .... no [(r)]
reversible — *from every reachable marking, there is a transition path going back to the initial marking* .... no [(s)]
live — *for every transition t, from every reachable marking, one can reach a marking in which t can fire* .... no [(t)]

[(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 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(f)] stated by CÆSAR.BDD version 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(g)] stated by CÆSAR.BDD version 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(h)] stated by CÆSAR.BDD version 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(i)] stated by CÆSAR.BDD version 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(j)] stated by CÆSAR.BDD version 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(k)] stated by CÆSAR.BDD version 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(l)] stated by PNML2NUPN 3.2.0 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(m)] stated by PNML2NUPN 3.2.0 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(n)] the definition of Nested-Unit Petri Nets (NUPN) is available from `http://mcc.lip6.fr/nupn.php`
[(o)] stated by CÆSAR.BDD version 3.7 on all 15 instances (3 types of models (Circular, Heawood, Kneser) × 5 instances).
[(p)] undead and survivor tokens can freely move into their own group.
[(q)] this is false when X and $Y \geq 1$, which is the case in all our instances.
[(r)] there are no deadlocks because a move transition can always fire if $L \geq 2$, which is the case in all our instances.
[(s)] It is not possible after the first occurrence of a kill transition, which is not dead when X and $Y \geq 1$ and is the case in all our instances.
[(t)] It is always possible to reach a configuration with 0 undead when $X \geq 1$, in which case the kill transition becomes dead. A similar observation holds for the bite transition when we have no more survivors.

Silvano Dal Zilio
dalzilio@laas.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
Circular (4, 5, 5)	29 812 [u]	336 112 [v]	10 [w]	10 [x]
Circular (4, 100, 100)	4.8483e+13	1.1045e+15	200 [y]	200 [z]
Circular (8, 100, 100)	3,6559e+23	1,5846e+25	200 [aa]	200 [ab]
Circular (32, 50, 50)	1.5644e+46	1.5919e+48	100 [ac]	100 [ad]
Circular (64, 10, 10)	1.6204e+24	5.9708e+25	20 [ae]	20 [af]
Heawood (14, 5, 5)	3.0397e+08	5.5687e+09	10 [ag]	10 [ah]
Heawood (14, 60, 60)	7,3066e+29	6,0032e+31	120 [ai]	120 [aj]
Heawood (17, 25, 25)	2,1698e+23	1,7974e+25	50 [ak]	50 [al]
Heawood (17, 50, 50)	4,1849e+31	4,5673e+33	100 [am]	100 [an]
Heawood (38, 15, 15)	1,3413e+26	7,3638e+27	30 [ao]	30 [ap]
Kneser (21, 7, 2, 5, 5)	1.0073e+11	8.2826e+12	10 [aq]	10 [ar]
Kneser (21, 7, 2, 60, 60)	7.4715e+38	2.4963e+41	120 [as]	120 [at]
Kneser (21, 7, 2, 80, 80)	2,7492e+43	9,8696e+45	160 [au]	160 [av]
Kneser (35, 7, 3, 60, 60)	6,7620e+52	1,3841e+55	120 [aw]	120 [ax]
Kneser (35, 7, 3, 100, 100)	?	?	200 [ay]	200 [az]

(u) computed by TINA version 3.8.5 on February 2025.
(v) computed by TINA version 3.8.5 on February 2025.
(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.
(aa) number of initial tokens, because the net is sub-conservative.
(ab) number of initial tokens, because the net is sub-conservative.
(ac) number of initial tokens, because the net is sub-conservative.
(ad) number of initial tokens, because the net is sub-conservative.
(ae) number of initial tokens, because the net is sub-conservative.
(af) number of initial tokens, because the net is sub-conservative.
(ag) number of initial tokens, because the net is sub-conservative.
(ah) number of initial tokens, because the net is sub-conservative.
(ai) number of initial tokens, because the net is sub-conservative.
(aj) number of initial tokens, because the net is sub-conservative.
(ak) number of initial tokens, because the net is sub-conservative.
(al) number of initial tokens, because the net is sub-conservative.
(am) number of initial tokens, because the net is sub-conservative.
(an) number of initial tokens, because the net is sub-conservative.
(ao) number of initial tokens, because the net is sub-conservative.
(ap) number of initial tokens, because the net is sub-conservative.
(aq) number of initial tokens, because the net is sub-conservative.
(ar) number of initial tokens, because the net is sub-conservative.
(as) number of initial tokens, because the net is sub-conservative.
(at) number of initial tokens, because the net is sub-conservative.
(au) number of initial tokens, because the net is sub-conservative.
(av) number of initial tokens, because the net is sub-conservative.
(aw) number of initial tokens, because the net is sub-conservative.
(ax) number of initial tokens, because the net is sub-conservative.
(ay) number of initial tokens, because the net is sub-conservative.
(az) number of initial tokens, because the net is sub-conservative.

Silvano Dal Zilio
dalzilio@laas.fr

Other properties

A winning configuration in the zombies and survivors game, depending on your perspective, is a marking in which there are no more undead (resp. humans). It is not possible to deal with optimal or winning *strategies* per se using only temporal logic, but we can express the fact that a winning condition can always be reached. This gives the following reachability and CTL formulas, where **undead** is the sum of markings in the places undead_ni for $i \in 1..L$, and similarly for **survivor**.

$$\text{WINNABLE} \;:\; \text{EF} \;\; (\textbf{undead} = 0)$$

$$\text{STRATEGY} \;:\; \text{AG} \;\; ((\textbf{survivor} \geq 1) \Longrightarrow \text{WINNABLE})$$

Note that a winning configuration is not a deadlock when the graph has at least two nodes ($L \geq 2$), since the camp that remains can always move. As a result, there are no deadlocks in the model. More interestingly, it is also not possible to reach a configuration with a single undead and no survivors (if we start with at least one survivor):

$$\text{ALONE} \;:\; \text{AG} \;\; ((\textbf{undead} = 1) \implies (\textbf{survivor} \geq 1))$$