Description

A Place-Transition net representing a variant of the Dekker’s mutual exclusion algorithm for \( N > 2 \) processes. Each process has three states, \( p_0 \), \( p_1 \), and \( p_3 \). \( p_0 \) is initial. From there, the process executes try and raises its flag, reaching \( p_1 \). In \( p_1 \), if at least one of the other process has a high flag, it withdraws its intent and goes back to \( p_0 \). In \( p_1 \), it enters the critical section if all other process’ flag is zero. From \( p_3 \), the process can only exit the critical section.

References

https://code.google.com/p/cunf/source/browse/tools/mkdekker.py

Scaling parameter

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Parameter description</th>
<th>Chosen parameter values</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N )</td>
<td>Number of processes</td>
<td>10, 15, 20, 50, 100, 200</td>
</tr>
</tbody>
</table>
Size of the model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of places</th>
<th>Number of transitions</th>
<th>Number of arcs</th>
</tr>
</thead>
<tbody>
<tr>
<td>N = 10</td>
<td>5N</td>
<td>N^2 + 2N</td>
<td>O(N^2)</td>
</tr>
<tr>
<td>N = 15</td>
<td>50</td>
<td>120</td>
<td>820</td>
</tr>
<tr>
<td>N = 20</td>
<td>75</td>
<td>255</td>
<td>1830</td>
</tr>
<tr>
<td>N = 50</td>
<td>100</td>
<td>440</td>
<td>3240</td>
</tr>
<tr>
<td>N = 100</td>
<td>250</td>
<td>2600</td>
<td>20100</td>
</tr>
<tr>
<td>N = 200</td>
<td>1000</td>
<td>40400</td>
<td>320400</td>
</tr>
</tbody>
</table>

Structural properties

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

Behavioural properties

- **safe** — in every reachable marking, there is no more than one token on a place
- **deadlock** — there exists a reachable marking from which no transition can be fired
- **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
- **live** — for every transition t, from every reachable marking, one can reach a marking in which t can fire

---

(a) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(b) transitions “enter_2” and “try_0” share a common input place “flag_0.0”, but only the former transition has input place “flag_0.1”.
(c) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(d) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(e) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(f) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(g) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(h) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(i) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(j) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(k) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(l) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(m) stated by CÆSAR.BDD version 1.7 on all 6 instances (10, 15, 20, 50, 100, and 200).
(n) the definition of Nested-Unit Petri Nets (NUPN) is available from http://mcc.lip6.fr/nupn.php
(o) stated by CÆSAR.BDD version 2.0 to be true on 3 instance(s) out of 6, and unknown on the remaining 3 instance(s).
(p) stated by CÆSAR.BDD version 2.0 to be false on 3 instance(s) out of 6, and unknown on the remaining 3 instance(s); confirmed at MCC’2014 by Tapaal, GreatSPN, and Lola on the 2, 3, and 4 smallest instances, respectively.
(q) stated by CÆSAR.BDD version 2.0 to be true on 3 instance(s) out of 6, and unknown on the remaining 3 instance(s).
Size of the marking graphs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of reachable markings</th>
<th>Number of transition firings</th>
<th>Max. number of tokens per place</th>
<th>Max. number of tokens per marking</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N = 10 )</td>
<td>6144 (^{(r)})</td>
<td>171,530 (^{(r)})</td>
<td>1 (^{(s)})</td>
<td>20 (^{(s)})</td>
</tr>
<tr>
<td>( N = 15 )</td>
<td>278,528 (^{(r)})</td>
<td>1.6835E+7 (^{(r)})</td>
<td>1 (^{(s)})</td>
<td>30 (^{(s)})</td>
</tr>
<tr>
<td>( N = 20 )</td>
<td>1.1534E+7 (^{(r)})</td>
<td>1.2164E+9 (^{(r)})</td>
<td>1 (^{(s)})</td>
<td>40 (^{(s)})</td>
</tr>
<tr>
<td>( N = 50 )</td>
<td>2.9273E+16 (^{(r)})</td>
<td>? (^{(r)})</td>
<td>1 (^{(s)})</td>
<td>100 (^{(s)})</td>
</tr>
<tr>
<td>( N = 100 )</td>
<td>6.4650E+31 (^{(r)})</td>
<td>? (^{(r)})</td>
<td>1 (^{(s)})</td>
<td>200 (^{(s)})</td>
</tr>
<tr>
<td>( N = 200 )</td>
<td>1.6230E+62 (^{(r)})</td>
<td>? (^{(r)})</td>
<td>1 (^{(s)})</td>
<td>400 (^{(s)})</td>
</tr>
</tbody>
</table>

Other properties

Mutual exclusion is guaranteed: no reachable marking covers any two places \( p_3/i, p_3/j \) with \( i \neq j \) and \( i, j \in \{1, \ldots, N\} \). Unfair runs are however possible.

---

\(^{(r)}\) computed at MCC’2013 by ITS-Tools, Marcie, Neco, and PNXDD; confirmed by CÆSAR.BDD version 1.8; confirmed at MCC’2014 by GreatSPN, Marcie, PNMC, PNXDD, Stratagem, and Tapaal.

\(^{(s)}\) computed at MCC’2014 by Marcie.

\(^{(t)}\) confirmed at MCC’2014 by GreatSPN, Marcie, PNMC, and Tapaal.

\(^{(u)}\) number of initial tokens, because the net is conservative.

\(^{(v)}\) computed at MCC’2013 by ITS-Tools, Marcie, Neco, and PNXDD; confirmed by CÆSAR.BDD version 1.8; confirmed at MCC’2014 by GreatSPN, Marcie, PNMC, PNXDD, Stratagem, and Tapaal.

\(^{(w)}\) computed at MCC’2014 by Marcie.

\(^{(x)}\) confirmed at MCC’2014 by GreatSPN, Marcie, PNMC, and Tapaal.

\(^{(y)}\) number of initial tokens, because the net is conservative.

\(^{(z)}\) computed at MCC’2013 by ITS-Tools, Marcie, Neco, and PNXDD; confirmed by CÆSAR.BDD version 1.8; confirmed at MCC’2014 by GreatSPN, Marcie, PNMC, and PNXDD.

\(^{(aa)}\) computed at MCC’2014 by Marcie.

\(^{(ab)}\) confirmed at MCC’2014 by GreatSPN, Marcie, and PNMC.

\(^{(ac)}\) number of initial tokens, because the net is conservative.

\(^{(ad)}\) computed at MCC’2014 by PNMC.

\(^{(ae)}\) computed at MCC’2014 by PNMC.

\(^{(af)}\) number of initial tokens, because the net is conservative.

\(^{(ag)}\) computed at MCC’2014 by PNMC.

\(^{(ah)}\) computed at MCC’2014 by PNMC.

\(^{(ai)}\) number of initial tokens, because the net is conservative.

\(^{(aj)}\) computed at MCC’2014 by PNMC.

\(^{(ak)}\) computed at MCC’2014 by PNMC.

\(^{(al)}\) number of initial tokens, because the net is conservative.