SAN Lite-Solver: a user-friendly software tool to solve SAN models

Afonso Sales
afonso.sales@pucrs.br

Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)

Porto Alegre, Brazil
Outline

- Stochastic Automata Networks (SAN)
- SAN Lite-Solver
- Performance Results
- Conclusion
STOCHASTIC AUTOMATA NETWORKS (SAN)
Stochastic Automata Networks (SAN) [Plateau 85]

- Structured Markovian formalism
- Describes in a structured manner a large system by its parts (automata)
- Each automaton is represented by states, events and related transitions
- Local events change the state of only one automaton
- Synchronizing events change the state of more than one automaton
- Input parameters (related to events) can be global system dependent (functions)
Stochastic Automata Networks (SAN)

- **Stochastic context**
  - Events are *stochastically* independent
  - Continuous time: *exponential* time between events occurrence
  - A SAN model describes a Markov Chain (*tensor algebra*)

- **Main objective**: cope with (large) systems composed of interacting components for quantitative analysis and prediction, and also property verification

- **Challenge**: handle the size explosion
  - with N automata: exponential growth of the state space
Exclusive communication between neighbor nodes (ECN)

- Set of nodes connected on a data bus
- A node can only communicate with its neighbors
- Exclusive communication: if two nodes are communicating, the other ones must wait
SAN model (ECN)

Product State Space (PSS) = 27
Reachable State Space (RSS) = 11

<table>
<thead>
<tr>
<th>Type</th>
<th>Event</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>loc</td>
<td>L1..L6</td>
<td>β1..β2</td>
</tr>
<tr>
<td>syn</td>
<td>S12</td>
<td>f12</td>
</tr>
<tr>
<td>syn</td>
<td>S21</td>
<td>f21</td>
</tr>
<tr>
<td>syn</td>
<td>S23</td>
<td>f23</td>
</tr>
<tr>
<td>syn</td>
<td>S32</td>
<td>f32</td>
</tr>
</tbody>
</table>

f12 = α12, if N3 is in state I
f21 = α21, if N3 is in state I
f23 = α23, if N1 is in state I
f32 = α32, if N1 is in state I
Solving SAN models

- Steady-state probability vector $\pi$ is computed using an iterative method (e.g., Power method):
  \[ \pi^{(i)} = \pi^{(i-1)} \times Q \]

- Markovian Descriptor:
  \[ Q = \sum_{j=1}^{N} \sum_{i=1}^{N} g \cdot Q_{j}^{(i)} + \sum_{e \in E_s} \left( \sum_{i=1}^{N} g \cdot Q_{e+}^{(i)} + \sum_{i=1}^{N} g \cdot Q_{e-}^{(i)} \right) \]
  where $Q_{j}^{(i)} = \begin{cases} Q_{i}^{(i)} & \text{if } j = i \\ I_{|S^{(i)}|} & \text{if } j \neq i \end{cases}$
The PEPS software tool

- Computes and tests the tensor formulation
- Implements several iterative methods:
  - Power
  - Jacobi
  - GMRES
- Allows transient analysis
- Computes performance indices
- **BUT** it works over all state space (*i.e.*, the model’s PSS)
Motivation

- Although structured formalisms provide a high-level abstraction, this modularity may lead to a large number of invalid combinations (states)
  - A structured model usually has $\text{RSS} \ll \text{PSS}$

### ECN model

<table>
<thead>
<tr>
<th>Nodes</th>
<th>PSS</th>
<th>RSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>243</td>
<td>19</td>
</tr>
<tr>
<td>10</td>
<td>59,049</td>
<td>39</td>
</tr>
<tr>
<td>15</td>
<td>14,348,907</td>
<td>59</td>
</tr>
<tr>
<td>20</td>
<td>3,486,784,401</td>
<td>79</td>
</tr>
<tr>
<td>$N$</td>
<td>$3^N$</td>
<td>$4N - 1$</td>
</tr>
</tbody>
</table>
Main goal

- It is to propose a software tool that computes the steady-state solution of a model in an easy and fast manner.
What is easy and difficult?

- **Easy**
  - Understand the structured description by automata
  - Understand the dynamics of events
  - Rules to generate the global system

- **Difficult**
  - Handle the possible global dependency of transitions
  - Generate efficiently the global description
SAN LITE-SOLVER
Overall solution scheme

SAN Lite-Solver

SAN model

SAN Lite-Solver

Compiler

Reachability Descriptor (transition system)

Initial state

underlying MC (HBF matrix)

Solution (iterative Power method)

probability vector

Results
Command-line software tool

\texttt{san-lite-solver <SAN model> [ options ]}

- Some options:
  - \texttt{-res} saves the results into a file
  - \texttt{-stt} saves all reachable states into a file
  - \texttt{-sim} saves the underlying MC of the model into a file (row-oriented type)
  - ...

WEB SITE

http://www.inf.pucrs.br/afonso.sales/san-lite-solver/
PERFORMANCE RESULTS
Models

- Wireless *ad hoc* Networks (WN)
- Kanban system (KS)
- Resource Sharing (RS)
- Dining Philosophers (DP)
- First Available Server (FAS)
## Performance results (memory)

<table>
<thead>
<tr>
<th>Model</th>
<th>PSS</th>
<th>RSS</th>
<th>RD</th>
<th>MDD</th>
<th>HBF</th>
<th>Total</th>
<th>PEPS DESCRIPTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>WN (N=16)</td>
<td>19,131,876</td>
<td>1,766</td>
<td>0.02 MB</td>
<td>0.03 MB</td>
<td>0.12 MB</td>
<td>0.17 MB</td>
<td>4.58 MB</td>
</tr>
<tr>
<td>WN (N=18)</td>
<td>172,186,884</td>
<td>4,622</td>
<td>0.03 MB</td>
<td>0.03 MB</td>
<td>0.33 MB</td>
<td>0.39 MB</td>
<td>41.07 MB</td>
</tr>
<tr>
<td>WN (N=20)</td>
<td>1,549,681,956</td>
<td>12,102</td>
<td>0.03 MB</td>
<td>0.03 MB</td>
<td>0.91 MB</td>
<td>0.97 MB</td>
<td>369.50 MB</td>
</tr>
<tr>
<td>KS (N=2)</td>
<td>531,441</td>
<td>4,600</td>
<td>0.01 MB</td>
<td>0.03 MB</td>
<td>0.53 MB</td>
<td>0.57 MB</td>
<td>4.23 MB</td>
</tr>
<tr>
<td>KS (N=3)</td>
<td>16,777,216</td>
<td>58,400</td>
<td>0.01 MB</td>
<td>0.11 MB</td>
<td>8.15 MB</td>
<td>8.27 MB</td>
<td>132.05 MB</td>
</tr>
<tr>
<td>KS (N=4)</td>
<td>244,140,625</td>
<td>454,475</td>
<td>0.01 MB</td>
<td>0.35 MB</td>
<td>71.13 MB</td>
<td>71.49 MB</td>
<td>1.88 GB</td>
</tr>
<tr>
<td>RS (P=14, R=11)</td>
<td>196,608</td>
<td>16,278</td>
<td>0.03 MB</td>
<td>0.55 MB</td>
<td>3.83 MB</td>
<td>4.41 MB</td>
<td>0.07 MB</td>
</tr>
<tr>
<td>RS (P=17, R=7)</td>
<td>1,048,576</td>
<td>41,226</td>
<td>0.04 MB</td>
<td>0.65 MB</td>
<td>8.67 MB</td>
<td>9.36 MB</td>
<td>0.27 MB</td>
</tr>
<tr>
<td>RS (P=20, R=5)</td>
<td>6,291,456</td>
<td>21,700</td>
<td>0.06 MB</td>
<td>0.69 MB</td>
<td>3.57 MB</td>
<td>4.32 MB</td>
<td>1.53 MB</td>
</tr>
<tr>
<td>DP (N=15)</td>
<td>14,384,907</td>
<td>470,832</td>
<td>0.03 MB</td>
<td>0.02 MB</td>
<td>79.70 MB</td>
<td>79.75 MB</td>
<td>3.44 MB</td>
</tr>
<tr>
<td>DP (N=16)</td>
<td>43,046,721</td>
<td>1,136,689</td>
<td>0.04 MB</td>
<td>0.02 MB</td>
<td>203.62 MB</td>
<td>203.68 MB</td>
<td>10.28 MB</td>
</tr>
<tr>
<td>DP (N=17)</td>
<td>129,140,163</td>
<td>2,744,210</td>
<td>0.04 MB</td>
<td>0.03 MB</td>
<td>518.64 MB</td>
<td>518.71 MB</td>
<td>30.81 MB</td>
</tr>
<tr>
<td>FAS (N=20)</td>
<td>1,048,576</td>
<td>1,048,576</td>
<td>0.03 MB</td>
<td>0.01 MB</td>
<td>200.00 MB</td>
<td>200.04 MB</td>
<td>0.27 MB</td>
</tr>
<tr>
<td>FAS (N=21)</td>
<td>2,097,152</td>
<td>2,097,152</td>
<td>0.03 MB</td>
<td>0.01 MB</td>
<td>416.00 MB</td>
<td>416.04 MB</td>
<td>0.52 MB</td>
</tr>
<tr>
<td>FAS (N=22)</td>
<td>4,194,304</td>
<td>4,194,304</td>
<td>0.04 MB</td>
<td>0.01 MB</td>
<td>864.00 MB</td>
<td>864.05 MB</td>
<td>1.02 MB</td>
</tr>
</tbody>
</table>
# Performance results (CPU time)

<table>
<thead>
<tr>
<th>Model</th>
<th>PSS</th>
<th>RSS</th>
<th>RD+MDD</th>
<th>HBF</th>
<th>Solution</th>
<th>Total</th>
<th>PEPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>WN (N=16)</td>
<td>19,131,876</td>
<td>1,766</td>
<td>≈0.00 s</td>
<td>0.03 s</td>
<td>1.65 s</td>
<td>1.68 s</td>
<td>1.76 day</td>
</tr>
<tr>
<td>WN (N=18)</td>
<td>172,186,884</td>
<td>4,622</td>
<td>≈0.00 s</td>
<td>0.08 s</td>
<td>6.54 s</td>
<td>6.62 s</td>
<td></td>
</tr>
<tr>
<td>WN (N=20)</td>
<td>1,549,681,956</td>
<td>12,102</td>
<td>≈0.00 s</td>
<td>0.24 s</td>
<td>19.85 s</td>
<td>20.09 s</td>
<td></td>
</tr>
<tr>
<td>KS (N=2)</td>
<td>531,441</td>
<td>4,600</td>
<td>≈0.00 s</td>
<td>0.06 s</td>
<td>0.09 s</td>
<td>0.15 s</td>
<td>298.89 s</td>
</tr>
<tr>
<td>KS (N=3)</td>
<td>16,777,216</td>
<td>58,400</td>
<td>≈0.00 s</td>
<td>0.88 s</td>
<td>2.12 s</td>
<td>3.00 s</td>
<td>16,201.98 s</td>
</tr>
<tr>
<td>KS (N=4)</td>
<td>244,140,625</td>
<td>454,475</td>
<td>0.01 s</td>
<td>7.52 s</td>
<td>24.72 s</td>
<td>32.25 s</td>
<td></td>
</tr>
<tr>
<td>RS (P=14, R=11)</td>
<td>196,608</td>
<td>16,278</td>
<td>0.01 s</td>
<td>0.86 s</td>
<td>0.09 s</td>
<td>0.96 s</td>
<td>8.06 s</td>
</tr>
<tr>
<td>RS (P=17, R=7)</td>
<td>1,048,576</td>
<td>41,226</td>
<td>0.01 s</td>
<td>2.93 s</td>
<td>0.15 s</td>
<td>3.09 s</td>
<td>55.65 s</td>
</tr>
<tr>
<td>RS (P=20, R=5)</td>
<td>6,291,456</td>
<td>21,700</td>
<td>0.01 s</td>
<td>1.84 s</td>
<td>0.03 s</td>
<td>1.88 s</td>
<td>223.06 s</td>
</tr>
<tr>
<td>DP (N=15)</td>
<td>14,384,907</td>
<td>470,832</td>
<td>≈0.00 s</td>
<td>11.63 s</td>
<td>20.03 s</td>
<td>31.66 s</td>
<td>5,026.45 s</td>
</tr>
<tr>
<td>DP (N=16)</td>
<td>43,046,721</td>
<td>1,136,689</td>
<td>≈0.00 s</td>
<td>30.47 s</td>
<td>55.54 s</td>
<td>86.01 s</td>
<td>16,860.77 s</td>
</tr>
<tr>
<td>DP (N=17)</td>
<td>129,140,163</td>
<td>2,744,210</td>
<td>≈0.00 s</td>
<td>79.47 s</td>
<td>150.74 s</td>
<td>230.21 s</td>
<td></td>
</tr>
<tr>
<td>FAS (N=20)</td>
<td>1,048,576</td>
<td>1,048,576</td>
<td>≈0.00 s</td>
<td>18.89 s</td>
<td>24.49 s</td>
<td>43.38 s</td>
<td>63.26 s</td>
</tr>
<tr>
<td>FAS (N=21)</td>
<td>2,097,152</td>
<td>2,097,152</td>
<td>≈0.00 s</td>
<td>40.45 s</td>
<td>55.31 s</td>
<td>95.76 s</td>
<td>138.44 s</td>
</tr>
<tr>
<td>FAS (N=22)</td>
<td>4,194,304</td>
<td>4,194,304</td>
<td>≈0.00 s</td>
<td>82.19 s</td>
<td>120.55 s</td>
<td>202.74 s</td>
<td>301.57 s</td>
</tr>
</tbody>
</table>
CONCLUSION
Conclusion

- **SAN Lite-Solver:**
  - Software tool to solve SAN models in a fast, easy and intuitive way
  - exploits the modeling power of the SAN formalism
  - works over the model’s RSS
    - spends more memory, but it is faster than PEPS
  - Exports the underlying Markov chain that can be used by other tools

- Limitation: hard to solve “fully connected” models

http://www.inf.pucrs.br/afonso.sales/san-lite-solver/
Future work

- Implement other solution methods:
  - Direct (e.g., Gaussian elimination)
  - Simulation
  - Uniformization (transient analysis)
SAN Lite-Solver: a user-friendly software tool to solve SAN models

http://www.inf.pucrs.br/afonso.sales/san-lite-solver/

Afonso Sales
afonso.sales@pucrs.br

Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)

Porto Alegre, Brazil