STOCHASTIC ACTIVITY NETWORKS

Morteza Golkari
Elham Arshad
Outline

• Introduction
• Activity Networks (AN)
• Activity Networks to Activity Systems
• Stochastic Activity Networks (SAN)
• Example
• Tools
Introduction

- Stochastic Activity Networks (SANs) are a convenient, graphical, high-level language for describing system behavior.
- SANs are a stochastic generalization of Petri nets which have been defined for the modeling and analysis of distributed real-time systems.
Introduction

- Stochastic Activity Networks (SANs) Have been used since the mid-1980s
- Prof. Movaghar presented a new definition for SANs for the first time
- More powerful and flexible than:
  - stochastic Petri nets
  - generalized stochastic Petri nets.
- This definition is based on a unified view of the system in three settings:
  - Nondeterministic
  - Probabilistic
  - Stochastic
Activity Network

- Activity networks are **nondeterministic** models which have been developed for representing **concurrent** systems.

- Informally, ANs are generalized Petri nets with the following primitives:
  - Places
  - Activities
    - Timed
    - Instantaneous
    - (case)
  - Input Gates: finite set of inputs and one output
  - Output Gates: finite set of outputs and one input
Formal Definition

\[ AN = (P, IA, TA, IG, OG, IR, OR) \]

- \( P \) is a finite set of places
- \( IA \) is a finite set of instantaneous
- \( TA \) is a finite set of timed activities
- \( IG \) is a finite set of input gates
- \( OG \) is a finite set of output gates
Formal Definition

\[ AN = (P, IA, TA, IG, OG, IR, OR) \]

- \( IR \subseteq P \times \{1, \ldots, |P|\} \times IG \times (IA \cup TA) \) is the input relation
- \( OR \subseteq (IA \cup TA) \times OG \times \{1, \ldots, |P|\} \times P \) is the output relation
Definitions

• Marking function
  • \( \mu : P \rightarrow \mathbb{N} \)

• Gates:
  • enabling function \( \rightarrow \) Boolean; also called the *enabling predicate*
  • Gate function \( \rightarrow \) state; rule for changing the state of the model

• Activity \( a \) is *enabled* in a marking \( \mu \)
  • For any input gate \( G \) of \( a \), all enabling predicates are true

• Marking \( \mu \) is *stable*
  • no instantaneous activity is enabled
Graphical Representation

- Places: $P_1, P_2, P_3, P_4, P_5$
- Timed Activities: $T_1, T_2, T_3$
- Instantaneous Activity: $I_1, I_2, I_3$
- Input Gates: $G_1, G_3, G_4$
- Output Gated: $G_2$
Graphical Representation

• **Input Relations:**
  • \((P_1, 1, G_3, I_1)\)
  • \((P_3, 2, G_3, I_1)\)

• **Output Relations:**
  • \((I_3, G_2, 1, P_1)\)
  • \((I_3, G_2, 2, P_2)\)

• **Marking:**
  • \((1, 2, 1, 0, 0)\)
Activity Network Behavior

- When there are more than one enabled TA or IA
  - Choice of which activity to complete first is done *nondeterministically*.

- *When activity complete, marking of the places may change*
  1. *Input gates function*
  2. *Output gates function*
Activity System

\[ AS = (Q, A, \rightarrow, Q_0) \]

- \( Q \) is a set of states
- \( A \) is the activity alphabet
- \( \rightarrow \subseteq Q \times A \times Q \) is the transition relation
- \( Q_0 \) is the set of initial states
Transformation of AN to AS (TS)

\[ (AN, \mu_0) \rightarrow AS = (Q, A, \rightarrow, Q_0) \]

- \( Q \) is the set of all stable markings of AN which are reachable from \( \mu_0 \)
- \( A = TA \)
- \( \mu, \mu' \in Q \) and \( a \in A, \mu \xrightarrow{a} \mu' \) iff
  - \( \mu' \) is reachable from \( \mu \) under a string of activities \( ax \)
    - \( a \in TA \)
    - \( x \) is a string of instantaneous activities or empty
- \( Q_0 \) is the set of all stable markings of AN which are reachable from \( \mu_0 \) under a (possibly an empty) string of instantaneous activities
SAN Formal Definition

\[ SAN = (P, IA, TA, IG, OG, IR, OR, C, F, \Pi, \rho) \]

• \( AN = (P, IA, TA, IG, OG, IR, OR) \)
• \( PAN = (P, IA, TA, IG, OG, IR, OR, C) \)
• \( C: \mathbb{N}^n \times IA \rightarrow [0,1] \) is the case probabilistic function, \( n = |P| \)
• \( F = \{ F(\cdot |\mu, a); \mu \in \mathbb{N}^n, a \in TA \} \) is the set of activity time distribution function
• \( \Pi: \mathbb{N}^n \times TA \rightarrow \{true, false\} \) is the reactivation predicate
• \( \rho: \mathbb{N}^n \times TA \rightarrow \mathbb{R}_+ \) is the enabling rate function
SAN Terms

- *Activation*: time at which an activity begins.
- *Completion*: time at which activity completes.
- *abort*: time after activation but before completion, when activity is no longer enabled.
- *active*: the time after an activity has been activated but before it completes or aborts.
SAN Terms

![Diagram showing SAN terms](image_url)
Mathematical Properties

• Same as Petri Nets
  • Reachability
  • Liveness
  • Boundedness
Example: **SAN Model of disks**
<table>
<thead>
<tr>
<th>Activity</th>
<th>Distribution</th>
<th>Parameter values</th>
</tr>
</thead>
<tbody>
<tr>
<td>disk_failure</td>
<td>exponential</td>
<td>rate $1/6000.0 \times \text{MARK(disks_working)}$</td>
</tr>
<tr>
<td>disk_repair</td>
<td>exponential</td>
<td>rate $\text{GLOBAL_D(repair_rate)}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gate</th>
<th>Definition</th>
</tr>
</thead>
</table>
| OGI  | $\text{MARK(disks\_failed)}++;$  
$\text{if (MARK(disks\_working) == 2)}$  
$\text{MARK(system\_fail)}++;$     |
| OG2  | $\text{MARK(disks\_working)}++;$  
$\text{if (MARK(disks\_working) == 3)}$  
$\text{MARK(system\_fail)} = -$; |
Tools For SAN Model

- METASAN
- UltraSAN
- Mobius
- SPNP
- SharifSAN
- SPTMAN
- Prism
SPTMAN

- Stochastic, Probabilistic, Timed Model Checker for Activity Networks
- In java, eclipse
- On XP/7/Vista/Linux
SPTMAN’s features

• Graphical modeling
• Using XML Format as text modeling
• SAN model checking by Continuous Stochastic Logic (CSL)
Xml Format

• Definition of Place:

<Place id="P1" name="state1" />
<InitialMarking val="4" />
</Place>
Xml Format (cont.)

• Definition of Time Activity:

```xml
<TimedActivity id="T1" name="TA1">
  <Rate val="1.25" />
  <!-<RateFunction val="func" />-->
  <EnablingRateFunction val="func"/>
</TimedActivity>
```
Xml Format (cont.)

• Definition of Instantaneous Activity:

  <InstantaneousActivity id="I1" name="IA1">
  <Probability val="1.25" />
  <!-- <ProbabilityFunction val="func" /> -->
  <InstantaneousActivity/>

Xml Format (cont.)

- Definition of Input Gate:

  
  ```xml
  <InputGate id="Ig1" name="Igate1">
  <InputFunction val="func"/>
  <EnablingPredicate val="func"/>
  <InputFunction/>
  ```
Xml Format (cont.)

- Definition of Output Gate:

```
<OutputGate id="Og1" name="Ogate1"/>
<OutputFunction val="func"/>
<OutputGate/>
```
Xml Format (cont.)

- Definition of Arc:

  <Arc id="arc1" source="P1" target="T1" />
• Definition of SAN Model:

  <SANML name="example" > ..... </SANML>
Prism

• Models:
  DTMC, CTMC, MDP, Probabilistic Automata

• Logic:
  CSL, PCTL, LTL
1 // single cell in wireless communication network [HHPT01]
2 // gsm/dsp 5/2/01
3
4 ctmc
5
6 const int N; // N - number of channels
7 const double lambdaAoA; // arrival rate of new calls
8 const double lambdaO2A; // arrival rate of hand-off calls
9 const double mu=1; // departure rate of calls
10
11 module cell
12
13   n : [0..N]; // number of calls in the cell
14
15   // arrival of new call
16   [] n\in[0,N) -> lambda_0 : (n'=n+1);
17
18   // arrival of hand of call
19   [] n=N -> lambda_0 : (n'=n+1);
20
21   // completion of call or mobile departs cell
22   [] n>0 -> n*mu : (n'=n-1);
23
24 endmodule
25
26 // rewards - number calls in the cell
27
28 rewards "calls"
29   true : n;
30
31 endrewards
Properties

P=? [ true U<=T (n=N) {n<N}(max) ]

P=? [ true U<=T (n=N^0.8) {n<N^0.8}(max) ]

P=? [ true U<=T (n=N^0.8) {n=N}(min) ]

R("call")=? [ i=T ]

S=? [ n<N^0.8 ]

R("call")=? [ S ]

The maximum probability that a handoff call can be dropped within t time units (assuming a guarded channel is free)

Constants

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>double</td>
<td></td>
</tr>
</tbody>
</table>
the maximum probability that a hand off call can be dropped within t time units (assuming a guarded channel is free)

- Constants

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>double</td>
<td></td>
</tr>
</tbody>
</table>
Property: \( P=? \{ \text{true} \land T (n=N) \{ n=N \}\}\text{max} \) 

Defined constants: 
\( N=20, \ T=6.08 \) 

Method: Verification 

Result: 
0.9999994520897092 (maximum value over states satisfying filter) 

The maximum probability that a hand off call can be dropped within time units (assuming a guarded channel is free)
QUESTION?

Thanks for Your Attention
References


• users.crhc.illinois.edu/nicol/ece541/slides