Using Integer Clocks to Verify Timing-Sync Sensor Network Protocol

Xiaowan Huang, Anu Singh, Scott Smolka
Stony Brook University
Introduction

- **Goal**
  - Model check the TPSN protocol.

- **Tool**
  - The Uppaal model checker for real-time systems modeled as networks of timed automata.

- **Problem**
  - Constraints of clocks in timed automata: a clock can only be assigned a constant value.
  - Requirement of arithmetic operations of clock values in the TPSN protocol.
TPSN Protocol

- To provide network-wide time synchronization in a wireless sensor network.

- Two steps
  - **Level-Discovering**: Establish a hierarchical structure in the network.
  - **Synchronization**: Perform pair wise synchronization along the edges of this structure.

- Result
  - All nodes in the network synchronize their clock with a reference node (the root).
Level-Discovering Phase

- Assign the root node level 0.
- The root broadcasts a `level_discovery` packet carrying its identity and level information to its immediate neighbors.
- The immediate neighbors of the root receive this packet and assign themselves a level, one greater than the level they have received.
- Repeat the process until every node in the network is assigned a level.
Synchronization Phase

Two way message exchange between a child (A) and its parent (B)

- T2 and T3 are measured in B's clock
- T1 and T4 are measured in A's clock

Diagram:

- Two-way exchange
- T2 and T3 are marked as "synchronization pulse"
- T1 and T4 are marked as "acknowledgement"
Synchronization Phase(2)

- Let $\Delta$ be the clock drift between $A$ and $B$.
- Let $d$ be the propagation delay.
- Assume $\Delta$ and $d$ do not change in a short time span.

- $T_2 - T_1 = d + \Delta \quad (1)$
- $T_4 - T_3 = d - \Delta \quad (2)$

  $\Rightarrow \Delta = \frac{(T_2 - T_4) - (T_1 - T_3)}{2}$

- $A$ performs clock adjustment: $t = t + \Delta$
The Uppaal Model Checker

- An integrated tool for specification, simulation and verification of real-time systems.
- Input of Uppaal: the XTA (eXtended Timed Automata) format.
- Clocks in Uppaal:
  - Can only be assigned an integer expression
  - Can only be compared with an integer expression or another clock
  - Clocks can not be read
  - Clocks can not advance by an arbitrary amount
To make clocks readable: use integer variables

- `typedef int intclock;`

To make clocks advance periodically: use a **Univiserval Pulse Generator** (UPG) process to broadcast a `time_pulse` signal to all processes in the system.
broadcast chan time_pulse;
process universal_pulse_generator()
{
    clock t;
    state S {t <= 1};
    init S;
    trans
    S -> S { guard t == 1;
        sync time_pulse!;
        assign t = 0; };
Processes deploying integer clocks.

For every state and every integer clock, the integer clock user process must specify a transition that responds to the `time_pulse` event (to advance the integer clock).

Drawbacks: an increase in model complexity, but the code to implement integer clocks is straightforward.
chan AtoB;
meta int msg;
process A()
{
    const int wait = 3;
    meta int clock x;
    state INIT, SENT;
    init INIT;
    trans
    INIT->INIT { sync time_pulse?; assign x=x+1; },
    SENT->SENT { sync time_pulse?; assign x=x+1; },
    INIT->SENT { guard x >= wait;
                 sync AtoB!; assign msg=x; }
}
Modeling TPSN

- Three Uppaal states are used for level-discovery phase and four are used for synchronization phase.

- Additional features
  - Time Drift and Resynchronization: node's clock has drifts from ideal clock. The whole network need to be resynchronized periodically before nodes' clocks diverge too much from the root.
  - Node dying/reviving.
Simulation Result

Time Drift

relative clock compared with root

root clock

n0
n1
n2
n3
n4
Verification Result

- No Deadlock: $A[] \text{ not deadlock}$
- Synchronized: $A[](\langle\rangle \text{ni.state == synchronized})$
- Relative Time Bounded: $A[] \text{ abs(ni.local_clock-n0.local_clock)< X}$
- Relative Time Close: $A[](\langle\rangle \text{abs(ni.local_clock-n0.local_clock)<Y})$
## Verification Result (2)

<table>
<thead>
<tr>
<th>Size of Network</th>
<th>No Deadlock</th>
<th>Synchronized</th>
<th>Relative Time Bounded</th>
<th>Relative Time Close</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.61 sec / 21 MB</td>
<td>2.06 sec / 24 MB</td>
<td>0.62 sec / 21 MB</td>
<td>2.11 sec / 24 MB</td>
</tr>
<tr>
<td>4</td>
<td>6.5 sec / 22 MB</td>
<td>68.0 sec / 31 MB</td>
<td>6.7 sec / 22 MB</td>
<td>70.2 sec / 31 MB</td>
</tr>
<tr>
<td>5</td>
<td>6.1 min / 126 MB</td>
<td>214.9 min / 181 MB</td>
<td>6.3 sec / 126 MB</td>
<td>236.4 min / 181 MB</td>
</tr>
</tbody>
</table>
Thank You!