Incremental Verification of Complex Event Processing Applications for System Monitoring

Andreas Metzger, Christian Reinartz, Klaus Pohl
Complex Event Processing

CEP Applications

• Collect, aggregate, analyse real-time event streams
• Detect patterns and generate more meaningful complex events
• Applied to monitor and detect problems during system execution; e.g.,
  • Physical machine monitoring in cloud systems
  • Tracking and tracing of transport processes

Verification of CEP Applications

• State space explosion problem!
  • Size of state space correlates with number of events to be checked
  • High number of events in CEP and high arrival rate
• Existing, dedicated CEP verification techniques
  • Limit verification to small subset of paths → verification results do not generalize
  • Set upper bounds on the number of events → unknown for which bound bug will show
Setting Upper Bounds

Bound\(_1\) is too low
\(\rightarrow\) Verification terminates before fault is reached

Bound\(_2\) is too high
\(\rightarrow\) Timeout reached before verification results are returned
Verification Approach

Incremental verification
• Verify model for incrementally larger bounds
• Iterate until time-out is reached

Naïve approach

Standard Model Checker
Verify entire model for each given bound

Advanced approach

Incremental Model Checker
Incrementally verify model for each new bound

Shortcoming
• Complete verification of model (redundant checks)

Two kinds of incremental model checkers
• Incrementally verify changed model
• Automatically increase state space (bounds)
Experiment

Performance data for naïve approach

- Non-incremental, bounded model checker Tapaal
- Model encoded as petri net [Reinartz et al. @ DEBS 2015]
- Measurements for each bound $b = 1, ..., 30$

Performance data for advanced approach

- Using benchmark results for bit vector model checker [Günther et al. @ SPIN 2014]
- Comparing incremental check of all bounds $(t_{inc})$ against non-incremental check of one bound $(t_{std})$ 

$$\lambda = \frac{(t_{inc} - t_{std})}{t_{std}}$$

- 32 benchmarks, 157 measurements

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>Percentage of measurements</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\leq 1$</td>
<td>30%</td>
</tr>
<tr>
<td>$\leq 2$</td>
<td>70%</td>
</tr>
<tr>
<td>$\leq 5$</td>
<td>80%</td>
</tr>
</tbody>
</table>
Experiment

Example CEP Application

Event Processing Agent

EPA 1: Load Truck

good A
Chemical Company Goods A

good B
Chemical Company Goods B

free truck
Truck Parking Space

EPA 2: Determine Target Warehouse

driving to warehouse 1

EPA 3: Arrival Warehouse 1

goods unloaded

driving to warehouse 1

EPA 4: Arrival Warehouse 2

goods unloaded

EPA 5: Transfer Warehouse 2 to Warehouse 1

warehouse limit exceeded

Warehouse 1

Warehouse 2

Incoming Event Stream

Outgoing Event Stream

Warehouse 2: Capacity 30 containers
- Transfer in case of overflow does not check safety property (only same kinds of critical goods may be stored at warehouse at any point in time)
Results

Verification Time [seconds]

Timeout

Naïve Approach

Advanced Approach

$\lambda = 5$

$\lambda = 2$

$\lambda = 1$

Up to $\lambda = 5$, advanced approach scales better!
Conclusion and Outlook

• Incremental approach for verifying CEP applications
• Detect faults in the CEP application during design time
  → Prevent false monitoring results at run time
• Advanced approach more scalable

• Future work
  • Prototypical implementation of advanced approach
    (adapting implementation of existing incremental model checkers to use EPNs as input)
Thank you!

Research leading to these results has received funding from...

...the EU’s Horizon 2020 research and innovation programme under under grant agreement no. 731932 http://www.transformingtransport.eu

...the EU’s Seventh Framework Programme FP7/2007-2013 under grant agreement 610802 http://www.cloudwave-fp7.eu/

...the EFRE co-financed operational program under grant agreement 005-1010-0012 http://www.lofip.de