Where did I go wrong?
Explaining errors in process models

Niels Lohmann
@nlohmann

Dirk Fahland
@dfahland
Verification of processes and services

process model

property

verification technique

diagnostic information
Verification of processes and services

BPMN

Soundness

domain-specific

verification technique

high-quality diagnostic information
Verification of processes and services

- WS-BPEL
- EPC
- YAWL
- Object Life-Cycles
- GSM
- CMMN
- BPMN
- Declare
- Rules
- WS-Policy
- Soundness
- Compliance
- Security
- Conformance to ...

- moving target
- domain-specific approaches too specific to follow
Verification of processes and services

WS-BPEL  EPC  YAWL
Object Life-Cycles  GSM
CMMN  BPMN  Declare
Rules
WS-Policy
Soundness
Compliance
Security
Conformance to ...

verification technique

general purpose

high-quality diagnostic information
Model checking

general purpose verification approach:

1. formalize model and specification*
2. push a button

* can be hidden from the user
Effectiveness and efficiency

- model checking works in reality
- successful applications in many domains

- very fast: “verify while you model”
Diagnosis

- in case of error: outputs target state and produce a witness path
- describes how target state can be reached
- operational semantics: can be simulated
Diagnosis: the bad

- paths can become very long
- length correlates with size of the model
- reports all events equally: disregarding importance
This talk: better diagnosis

Why useless?

essential

path
Reasons for useless paths

- detours
  - depth-first search
- indisputable parts
  - bootstrapping
- interleavings
  - concurrency
Running example

lack of synchronization
Reduction: obvious parts
- classify transitions
- only report points of alternative continuations*

* XOR-gateways, events, exceptions, … assume progress of flow
Non-obvious “core” of a path $\approx 10$-25%

<table>
<thead>
<tr>
<th>library</th>
<th>A</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>avg. path length before / after</td>
<td>17.51 / 1.83</td>
<td>17.52 / 2.11</td>
<td>16.06 / 1.54</td>
<td>20.34 / 1.67</td>
<td>13.40 / 2.30</td>
</tr>
<tr>
<td>max. path length before / after</td>
<td>53 / 8</td>
<td>66 / 7</td>
<td>56 / 6</td>
<td>54 / 5</td>
<td>21 / 3</td>
</tr>
<tr>
<td>sum of path lengths before / after</td>
<td>1699 / 178</td>
<td>1419 / 171</td>
<td>1349 / 129</td>
<td>1688 / 139</td>
<td>134 / 23</td>
</tr>
<tr>
<td>reduction</td>
<td>89.52 %</td>
<td>87.95 %</td>
<td>90.44 %</td>
<td>91.77 %</td>
<td>82.84 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>library</th>
<th>A</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>avg. path length before / after</td>
<td>30.83 / 3.17</td>
<td>10.47 / 0.66</td>
<td>12.16 / 0.68</td>
<td>11.50 / 0.59</td>
<td>51.00 / 7.57</td>
</tr>
<tr>
<td>max. path length before / after</td>
<td>89 / 13</td>
<td>52 / 7</td>
<td>100 / 8</td>
<td>103 / 14</td>
<td>120 / 17</td>
</tr>
<tr>
<td>sum of path lengths before / after</td>
<td>1079 / 111</td>
<td>1047 / 66</td>
<td>1459 / 82</td>
<td>1507 / 77</td>
<td>357 / 53</td>
</tr>
<tr>
<td>reduction</td>
<td>89.71 %</td>
<td>93.70 %</td>
<td>94.38 %</td>
<td>94.89 %</td>
<td>85.15 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>library</th>
<th>A</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>avg. path length before / after</td>
<td>12.06 / 2.79</td>
<td>13.82 / 2.55</td>
<td>18.13 / 2.33</td>
<td>14.27 / 2.55</td>
<td>11.27 / 2.33</td>
</tr>
<tr>
<td>max. path length before / after</td>
<td>44 / 7</td>
<td>70 / 7</td>
<td>95 / 7</td>
<td>95 / 7</td>
<td>27 / 3</td>
</tr>
<tr>
<td>sum of path lengths before / after</td>
<td>19699 / 4557</td>
<td>5707 / 1054</td>
<td>13835 / 1777</td>
<td>17494 / 3130</td>
<td>169 / 35</td>
</tr>
<tr>
<td>reduction</td>
<td>76.87 %</td>
<td>81.53 %</td>
<td>87.16 %</td>
<td>82.11 %</td>
<td>79.29 %</td>
</tr>
</tbody>
</table>
Reduction: spurious decisions

- can be found by **model checking**
- results: **50%-80% spurious**, occasionally no reduction (timeout)
Reasons for useless paths

- detours
- depth-first search

- indisputable parts
- bootstrapping

- interleavings
- concurrency
Reduction: unorder steps

- idea: show independence of steps
  (→ partially ordered runs)
- makes synchronization points (milestones) explicit

independent steps

many paths to same goal state

order of steps irrelevant
Reduction: unordered steps
More aid: preserve reference points
Final: remove obvious/spurious parts
Essential path: find source of error
Results: typical reduced paths

improper completion

2x lack of synchronization

deadlock
Summary

- general purpose verification more user friendly
- paths $\rightarrow$ partial order of important decisions
- applicable to any verification goal
- keep reference points to aid diagnosis

Next steps

- error localization vs. explanation
- detect useless cycles
- How should a good diagnosis for $problem$ look like?
Where did I go wrong?
Explaining errors in process models

Niels Lohmann

Universität Rostock
Traditio et Innovatio