An Analysis of the Cost of Validating Semantic Composability

Claudia Szabo** and Yong Meng Teo
Department of Computer Science
National University of Singapore
email: teoym@comp.nus.edu.sg

** Department of Computer Science
University of Adelaide, Australia
Prior Work


Outline

• Motivation

• Related Work

• Cost Analysis
  ▪ Methodology
  ▪ General Model Properties
  ▪ Model Execution
  ▪ Fork & Join

• Summary
What is Composability?

• “Capability to select and assemble simulation components in various combinations to satisfy user requirements”

• Two main levels of composability:
  
  ▪ syntactic – interoperation, data compatibility, …
  
  ▪ **semantic** – meaningful behavior, understanding of assumptions, context, … => focus of this paper
Validation of Semantic Composability

• Question: Does the composed model produce semantically correct results?

• Some issues:
  ▪ Semantic composability is not a closed operation
  ▪ Different validation perspectives:
    • Model properties
    • Model execution
    • ...
  ▪ Trade-off: cost vs credibility (validation accuracy)
  ▪ ...

Motivation

• Understand cost of validation and trade-off

• Different validation approaches:
  ▪ General model properties: I/O transformations, …
  ▪ Model execution: comparison with reference models, timeless vs time-based, ..

• Factors:
  ▪ Models: #components, #states, …
  ▪ Complexity of models: logic, communication, structure, ….
Related Work

  - Validation cost: 5-17.5\% of total M&S budget

- Industry reports [27]
  - Validation cost: 5-19\% of total M&S budget

- Balci and Sargent [26]
  - Relation between validation

![Graph showing the relationship between validation cost and model credibility/validation accuracy.](image-url)
Cost of Semantic Validation

• Factors:
  ▪ model size: #components, #attributes, #states, …
  ▪ composition structure: degree of component interaction
  ▪ …

▪ Validation Approach
  • validation techniques: model checking, bisimulation, …
  • levels of abstraction: timeless vs time; blackbox vs whitebox, …
## Comparison of Validation Approaches

<table>
<thead>
<tr>
<th>STEPS</th>
<th>APPROACHES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BOM [2007,2009]</td>
</tr>
<tr>
<td></td>
<td>DEVS [2006,2007]</td>
</tr>
<tr>
<td></td>
<td>Petty &amp; Weisel [2004]</td>
</tr>
<tr>
<td></td>
<td>Deny-validity [2009]</td>
</tr>
<tr>
<td>General Model Properties</td>
<td></td>
</tr>
<tr>
<td>1. Component Communication</td>
<td>event syntax</td>
</tr>
<tr>
<td>2. Component Coordination</td>
<td>Z-based DEVS</td>
</tr>
<tr>
<td>3. Component Computation</td>
<td>rule engine CD++ -based DEVS</td>
</tr>
<tr>
<td>Model Execution Validation</td>
<td>sequence</td>
</tr>
<tr>
<td></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>timeless</td>
</tr>
<tr>
<td></td>
<td>time-based</td>
</tr>
</tbody>
</table>

Semantic data compatibility

Timeless execution

Time-based execution
Four Validation Approaches

- **Z-based DEVS [9]**
  - Transform DEVS model into Z specification
  - Validate component coordination (specification errors)
  - Formal; *No synchronization*, not suited for coupled models

- **CD++ DEVS [10]**
  - DEVS coupled model: black-box
  - Validate component computation (input/output transformation)
  - Easy to use; *only primitive data types*

- **Petty & Weisel [2]**
  - Formal theory of composability, mathematical composition
  - Model execution validation: comparison with reference model
  - Reason about composability; *timeless* (no fork/join, …)

- **Deny-validity [11]**
  - Validate general model properties + model execution validation
  - Layered process with increasing accuracy at incremental cost; *time-based, fork & join*
Some Questions

• What is the cost of validating general model properties?
  ▪ CD++ DEVS, Z DEVS, Deny-validity

• What is the cost of model execution validation?
  ▪ Petty & Weisel, Deny-validity

• What is the trade-off between credibility and cost?
  ▪ Petty & Weisel, Deny-validity

• .........
## General Model Properties

Component communication (P1); Component Coordination (P2); Component Computation (P3)

<table>
<thead>
<tr>
<th>#Comp</th>
<th>Z DEVS P2</th>
<th>CD++</th>
<th>Deny-validity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DEVS P3</td>
<td>P1</td>
<td>P2</td>
</tr>
<tr>
<td>100</td>
<td>&lt;0.1</td>
<td>0.5</td>
<td>146.6</td>
</tr>
<tr>
<td>500</td>
<td>0.2</td>
<td>4.5</td>
<td>193.6</td>
</tr>
<tr>
<td>1,000</td>
<td>0.7</td>
<td>16.7</td>
<td>330.4</td>
</tr>
</tbody>
</table>

- Reduced runtime
- Type and syntax checking
- No interleaved execution states

- Increased runtime
  - Type, safety, liveness, deadlock freedom
  - All possible interleaved execution states

15 June 2011 25th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation
# General Model Properties

Component communication (P1); Component Coordination (P2); Component Computation (P3)

<table>
<thead>
<tr>
<th>#Comp</th>
<th>Z</th>
<th>CD++ DEVS P2</th>
<th>DEVS P3</th>
<th>Deny-validity</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>P1</td>
<td>P2</td>
</tr>
<tr>
<td>100</td>
<td>&lt;0.1</td>
<td>0.5</td>
<td></td>
<td>0.2</td>
<td>146.6</td>
</tr>
<tr>
<td>500</td>
<td>0.2</td>
<td>4.5</td>
<td></td>
<td>0.3</td>
<td>193.6</td>
</tr>
<tr>
<td>1,000</td>
<td>0.7</td>
<td>16.7</td>
<td></td>
<td>0.5</td>
<td>330.4</td>
</tr>
</tbody>
</table>

- Check types of data at connection points
- Last connection point
- Primitive types

- Check types of data at connection points + safety & liveness properties
- Any connection point
- Any data type (ontology)
Model Execution Validation

- 3-step process
  - Transform components to formalism (static vs time-based)
  - Transform composition into Labeled Transition System (LTS)
  - Compare with reference model (two LTS)

- Formalism
  - Timeless (Petty & Weisel) – reason about composability, no fork & join
  - Time-based (Deny-validity) – fork & join, costly
Timeless vs Time-based

Petty & Weisel composed model – no time, order based on position

Deny-validity – time, order based on time
Comparison with Reference Model

**Composed Model**
1. Formal Representation
   - $C_1$
   - ...

2. Unfolding and Sampling
3. Composition
4. Simulation

**Reference Model**
1. Formal Representation
   - $C_1$
   - ...

2. Unfolding and Sampling
3. Composition
4. Simulation

$L(M) \rightarrow L(M^*)$

**Validation**

- Strong equivalence?
  - Yes: **Valid**
  - No

- $L(M) \lor L(M^*)$?
  - Yes: **Valid**
  - No: **Invalid**
# Cost Factors

<table>
<thead>
<tr>
<th>Cost Factors</th>
<th>Petty &amp; Weisel</th>
<th>Deny-validity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formalism - ( f )</td>
<td>( \geq f(n, \tau) )</td>
<td>( f(n, s, t, \tau) )</td>
</tr>
<tr>
<td>Process - ( p )</td>
<td>( p(n, \tau) )</td>
<td>( p(n, s, t, \tau) )</td>
</tr>
<tr>
<td>Comparison - ( c )</td>
<td>( c(n, \tau) )</td>
<td>( c(n, \tau) + c.(n, a, \tau) )</td>
</tr>
</tbody>
</table>

- \( n \) – number of components
- \( \tau \) – validation window size; indirect measure of accuracy
- \( s \) – average number of states/component
- \( t \) – time
- \( a \) – average number of attributes per component
Experiment Setup

• Model Validation
  ▪ Deny-validity approach
    • Java implementation
    • Choco constraint solver – for time-based ordering
    • CADP bisimulation tool – for reference model comparison
  ▪ Petty & Weisel
    • Java implementation
    • CADP bisimulation tool – for reference model

• Platform
  ▪ Dell PowerEdge SC1430 Dual Quad Core Server, Intel Xeon, 1.83 GHz, 4GB RAM
Number of Components
Number of Components

\[ \tau = 3 \]
Validation Window Size

$\tau = 25$

$= 25 \times 1,000$ states
Composition Structure: Effects of Fork and Join

No Fork & Join

10% Fork & Join
Summary

- Cost of semantic validation significantly higher than initially envisaged

- Validation cost:
  - 1 second (single property) to 7 minutes (more comprehensive validation)
  - Composition structure: \( \uparrow 10\% \) fork & join \( \Rightarrow \) 50\%\( \uparrow \)
  - Time-based \( \Rightarrow \) high validation cost

- Trade-off: \( \uparrow 25\% \) validation window size (model credibility) \( \Rightarrow \) 5x cost
Publications
www.comp.nus.edu.sg/~teoym


Thank you
This paper

Composition Structure: without fork & join

![Diagram](image)

Component Coordination
Meta-Simulation
Model Execution Validation

![Bar Chart](image)

- Runtime (s)
- Composed Models
- (100, 1)
- (100, 10)
- (500, 3)
- (500, 10)
- (1,000, 3)
- (1,000, 10)
- (1,000, 20)
Composition Structure: 10% fork & join
Composition structure: without fork & join
Composition structure: 10% fork & join
Composition structure: without fork & join

Composition structure: 10% fork & join
Validation Window Size

\[ n = 1,000 \]

\[ 25 \times 1,000 \]