Partially Bounded Context-Aware Verification

LE ROUX Luka & TEODOROV Ciprian
Lab-STICC, ENSTA Bretagne, Brest, France
Introduction
Model-Checking

- Major algorithmic breakthroughs [ClarkeEmersonSifakis09]
  - Symbolic approach (OBDDs)
  - Partial order reduction
  - Bounded Model Checking
  - Abstraction Refinement Loop (CEGAR)

- When scalability issues persist
  - Refine the specifications
  - Narrow the modeling scope
  - Split the analysis

Exhaustive and automatic formal method [ClarkeEmerson82, QueilleSifakis82]
Introduction

Splitting the analysis
Case Study
Landing Gear System [F. Boniol, V. Wiels, ABZ’2014]
Context-Aware Verification [STTT’17]

LGS Specification

Interaction Alphabet

\[ A = \{ \text{handle, } f_1, \ldots, f_n \} \]

Guide

\[ G_1 \quad G_2 \quad \ldots \quad G_n \]

Verification Guide

Model

Labeled Transition System

Guide

\[ M \]

\[ P \]

\[ \text{All } G_i \quad M \quad P \]

\[ \text{Guide} \quad M \quad P \]
**xGDL**

**Operators**

<table>
<thead>
<tr>
<th>a</th>
<th>Interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>↓</td>
<td>Empty term</td>
</tr>
<tr>
<td>$C_1 ; C_2$</td>
<td>Sequence</td>
</tr>
<tr>
<td>$C_1 □ C_2$</td>
<td>Alternative</td>
</tr>
<tr>
<td>$C ?$</td>
<td>Optional</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$C *$</th>
<th>Repetition (0+)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C +$</td>
<td>Repetition (1+)</td>
</tr>
<tr>
<td>$C {i, j}$</td>
<td>Repetition (bounded)</td>
</tr>
<tr>
<td>$C_1</td>
<td></td>
</tr>
<tr>
<td>${i, j} \text{ of } [C_1, C_2, \ldots, C_n]$</td>
<td>Permutations</td>
</tr>
</tbody>
</table>

**Interaction Alphabet**

$A = \{ \text{Handle, } f_1, \ldots, f_n \}$

---

**Examples**

<table>
<thead>
<tr>
<th>Pilot</th>
<th>handle *</th>
<th>« Handle the landing gears at will »</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failures</td>
<td>${0, 3} \text{ of } [f_1, f_2, \ldots, f_{12}]$</td>
<td>« 0 to 3 unique failures among a set of 12 »</td>
</tr>
<tr>
<td>Guide</td>
<td>Pilot $</td>
<td></td>
</tr>
</tbody>
</table>
xGDL
Compilation

xGDL expression → NFA → DFA → xGDL Guide

Semantics

\[
\begin{align*}
C_1 \xrightarrow{a} C'_1 \\
C_1 \parallel C_2 \xrightarrow{a} C'_1 \parallel C_2 \\
\perp \parallel C \xrightarrow{T} C \\
C_1 \parallel C_2 \xrightarrow{a} C'_1 \parallel C'_2 \\
C_1 \parallel \perp \xrightarrow{T} C
\end{align*}
\]

\( handle^* \parallel \{0, 3\} \text{ of } [f_i, f_j, f_k] \)
xGDL Composition

Initial states \( G_0 \times S_0 \)

Synchronisation \( a \neq \tau, \ (g, s) \xrightarrow{a} (g', s') \iff g \xrightarrow{a} g' \land s \xrightarrow{a} s' \)

Stuttering steps \( (g, s) \xrightarrow{\tau} (g', s') \iff g = g' \land s \xrightarrow{\tau} s' \)

Always possible to produce a « neutral element »

\[ A = \{a_1, \ldots, a_n\}, \ G_{neutral} = (a_1 \Box \ldots \Box a_n)^* \]
Initial Guide
Production & Soundness

LGS Requirements

[...] Failures are irreversible
[...] Four or more failures is outside the scope

\[ G_{neutral} = (\text{handle} \square f_1 \square \ldots \square f_n)^* \]
\[ = \text{handle}^* \parallel (f_1 \square \ldots \square f_n)^* \]
\[ G_{scope} = \text{handle}^* \parallel \{0, n\} of [f_1, \ldots, f_n] \ (uniqueness) \]
\[ G_{scope} = \text{handle}^* \parallel \{0, 3\} of [f_1, \ldots, f_n] \ (at \ most \ 3) \]
Splitting the analysis
Illustration

\[ G_{\text{scope}} = \text{handle} \ast \ || \ \{0, 3\} \text{of} \ [f_1, \ldots, f_n] \]

⚠️ At most three failures may happen in one execution.
There are 720 distinct subsets of three failures.

\[ G_{\text{id}}^3 = \text{handle} \ast \ || \ \{0, 3\} \text{of} \ [f_i, f_j, f_k] \]

\[ \text{language}(G_{\text{scope}}) = \bigcup_{id=0}^{719} \text{language}(G_{\text{id}}^3) \]
Partially Bounded
Unrolling the guide

\[ G_{id} = handle^* \mid \{0, 3\} \mid [f_i, f_j, f_k] \]

DAG specific algorithms from CaV literature
- Split: an automatic, recursive decomposition
- PastFree[ze]: reduces memory load
Partially Bounded

Soundness

\[ G^1_{id} = \text{handle} \ast \bigparallel f_i \]

Resulting state space (indexed):

\[ \text{Unroll}(G^1_{id}, 35) \]

Model-Checking

\[ G^1_{id} \bigparallel M \bigparallel P \]

\[ 128\text{Gb} \]

\[ \text{Failure} | f_1, f_2, f_3, \ldots, f_{12} \]
\[ \text{Bound} | 16, 18, 17, 20, 20, X, X, 20, 20 \]

\[ \text{Table 2. Unrolling bounds required for completeness} \]
Conclusion

Interaction Alphabet
\[ A = \{ \text{Handle}, f1, \ldots, fn \} \]

\[ \text{Unroll}(G, n) \]
& Resulting state space analysis

\[ G \]
\[ M \]
\[ P \]

All
\[ G_{id} \]
\[ G_{scope} \]
\[ G_{0} \]
\[ G_{1} \]
\[ \ldots \]
\[ G_{719} \]

\[ G_{scope} \]
Future Works

• PastFree[ze] with DFAs (cycles)

• Tooling / automation of the induced state clusters bi-simulation

• Usage in a collective and heterogeneous verification task
Tusen takk!

(thank you!)

Questions