EXPTIME-complete Decision Problems for Modal and Mixed Specifications

Adam Antonik, Imperial College, London
Michael Huth, Imperial College, London
Kim G. Larsen, Aalborg University
Ulrik Nyman, Aalborg University
Andrzej Wąsowski, IT University of Copenhagen

EXPRESS 2008 Toronto, Canada
We Ask Complexity Questions For

**CI** Common Implementation

\[ S_1 \quad \text{or} \quad S_1 \cap S_2 \quad ? \]
We Ask Complexity Questions For

CI  Common Implementation

S_1  S_2  or  S_1 \cap S_2  ?

C  Consistency

S = \emptyset  or  . S  ?
We Ask Complexity Questions For

CI  Common Implementation

\[ S_1 \quad \text{or} \quad S_1 \cap S_2 \quad ? \]

C  Consistency

\[ S = \emptyset \quad \text{or} \quad \bullet \quad S \quad ? \]

TR  Thorough Refinement

\[ S_2 \quad \text{or} \quad S_2 \cap S_1 \quad ? \]
Agenda

• Modal and Mixed **Specifications** in a Nutshell
• The Problems and Our **Claims**
• Some **Proof** Sketches
• Open Issues & **Summary**
Part I

Modal & Mixed Specifications in A Nutshell
Some traces of the coffeemaker:

- insert coin, get coffee
- insert coin, get tea
- press cream, insert coin, get café au lait
An LTS + simulation refinement

- Overapproximate possible behaviors in each state
- An empty LTS "•" is a perfect refinement.
Modal Specifications
Larsen & Thomsen, LICS’88

• Under- and over-approximate behavior
• Each implementation **must** accept coins and produce coffee
• Cream or tea optional
• If cream offered then caffe-au-lait must be delivered

All required behavior (**must**) is allowed (**may**).
Refinement

May refines to must, may or nothing. Must refines to must.
May refines to must, may or nothing. Must refines to must.
Refinement

\[ S: \]

\[ T: \]

Infinitely many more refinements exist!!!
Refinement

$S:\quad \xrightarrow{\text{coin}}$

$T:\quad \xrightarrow{\text{coffee}} \xrightarrow{\text{cream}} \xrightarrow{\text{tea}} \xrightarrow{\text{caf\text'-au-lait}}$

But this is not a refinement!
A relation $\leq$ is refinement iff for every $s \leq t$ it holds that whenever $s \xrightarrow{a} s'$ then also $t \xrightarrow{a} t'$ for some $t'$ and $s' \leq t'$ whenever $t \xrightarrow{a} t'$ then also $s \xrightarrow{a} s'$ for some $s'$ and $s' \leq t'$
Implementations

\[ I: \]

\[ S: \]

\[ I \text{ is an implementation of } S \text{ iff} \]

\[ I \leq S \text{ and } \rightarrow_I = --\rightarrow_I \]
Mixed vs Modal Specifications

- **Modal** specifications: $\subseteq$  
  $\rightarrow$ Always have implementations (consistent)

- **Mixed** specifications: possibly $\not\subseteq$  
  $\rightarrow$ Larsen’89, Dams’96

- A **consistent** mixed specification:

- An **inconsistent** mixed specification:
Why Modal & Mixed Specifications?

- **Semantic foundation** for specification & verification
- Same spec **combines** under- & over-approximations → existential and universal properties in static analysis
- Refinement is the **mid-way** between simulation (too weak) & bisimulation (too strong)
- See **recent survey** by the authors for more applications and more results → Bulletin of EATCS, June 2008
Part II

The Problems &
Our Claims
Common Implementation

Problem CI

For modal (mixed) specifications $S_1$ and $S_2$ decide if

$$\exists \text{ implementation } I. \ I \leq S_1 \text{ and } I \leq S_2$$

Claim: EXPTIME-complete
Problem C

For a mixed specification $S$ decide if

\[ \exists \text{ implementation } I. \ I \leq S \]

\[ S = \emptyset \quad \text{or} \quad \bullet S \quad \text{?} \]

Claim: EXPTIME-complete

Remark: this problem is trivial for modal specifications.
Thorough Refinement

**Problem TR**

For a mixed specifications $S_1$ and $S_2$ decide if

$$\forall \text{ implementations } I. \ I \leq S_1 \text{ implies } I \leq S_2$$

**Claim:** EXPTIME-complete

**Remark:** this problem is open for modal specifications.
Note that refinement is in $P$, while TR is EXPTIME-complete. So Refinement and TR do not coincide.

(Hüttel’88) proves this using a counterexample in this spirit:

Implementations sets of $M$ and $N$ are equal, but $M \not\leq N$. Similar examples exist for properly modal specifications.
Part III

Proof Sketches
# Bounds Before This Work

Antonik et al. FOSSACS’08

<table>
<thead>
<tr>
<th></th>
<th>Modal spec.</th>
<th>Mixed spec.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CI</strong></td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td><strong>C</strong></td>
<td>trivial</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td><strong>TR</strong></td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
</tbody>
</table>

**FOSSACS’08:**
- Two complicated reductions showing the red `!'s.
- A chain of reductions along the red arrows.
## Bounds Before This Work

Antonik et al. FOSSACS’08

<table>
<thead>
<tr>
<th></th>
<th>Modal spec.</th>
<th>Mixed spec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>C</td>
<td>trivial</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>TR</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
</tbody>
</table>

**FOSSACS’08:**

- Two complicated reductions showing the red ¡’s.
- A chain of reductions along the red arrows.
## Bounds Before This Work

Antonik et al. FOSSACS’08

<table>
<thead>
<tr>
<th></th>
<th>Modal spec.</th>
<th>Mixed spec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>C</td>
<td>trivial</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>TR</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
</tbody>
</table>

**FOSSACS’08:**

- Two complicated reductions showing the red `!`'s.
- A chain of reductions along the red arrows.
### New Bounds — The Proof Structure

<table>
<thead>
<tr>
<th></th>
<th>Modal spec.</th>
<th>Mixed spec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>C</td>
<td>trivial</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>TR</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
</tbody>
</table>

- Prove hardness of CI for modal specifications
- By the know sequence of reductions arrive at the remaining results
- So far failed to reduce TR in the modal case
## New Bounds — The Proof Structure

<table>
<thead>
<tr>
<th></th>
<th>Modal spec.</th>
<th>Mixed spec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>EXPTIME-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>C</td>
<td>trivial</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>TR</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
</tbody>
</table>

- Prove hardness of CI for modal specifications
- By the known sequence of reductions arrive at the remaining results
- So far failed to reduce TR in the modal case
**New Bounds — The Proof Structure**

<table>
<thead>
<tr>
<th></th>
<th>Modal spec.</th>
<th>Mixed spec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>EXPTIME-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>C</td>
<td>trivial</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
<tr>
<td>TR</td>
<td>PSPACE-hard, EXPTIME</td>
<td>PSPACE-hard, EXPTIME</td>
</tr>
</tbody>
</table>

- Prove hardness of CI for modal specifications
- By the known sequence of reductions arrive at the remaining results
- So far failed to reduce TR in the modal case
### New Bounds — The Proof Structure

<table>
<thead>
<tr>
<th></th>
<th>Modal spec.</th>
<th>Mixed spec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>EXPTIME-hard, EXPTIME</td>
<td>EXPTIME-hard, EXPTIME</td>
</tr>
<tr>
<td>C</td>
<td>trivial</td>
<td>EXPTIME-hard, EXPTIME</td>
</tr>
<tr>
<td>TR</td>
<td>PSPACE-hard, EXPTIME</td>
<td>EXPTIME-hard, EXPTIME</td>
</tr>
</tbody>
</table>

- Prove hardness of CI for modal specifications
- By the known sequence of reductions arrive at the remaining results
- So far failed to reduce TR in the modal case
CI for Modal Specs is EXPTIME-complete

Most of the paper is devoted to EXPTIME-completeness of CI for Modal Specifications

The proof is by reduction from the acceptance problem for linearly bounded alternating Turing machines.

A teaser:

More in the paper.
## Summary

<table>
<thead>
<tr>
<th></th>
<th>Modal specifications</th>
<th>Mixed specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td>EXPTIME-complete</td>
<td>EXPTIME-complete</td>
</tr>
<tr>
<td>C</td>
<td>trivial</td>
<td>EXPTIME-complete</td>
</tr>
<tr>
<td>TR</td>
<td>PSPACE-hard, EXPTIME</td>
<td>EXPTIME-complete</td>
</tr>
</tbody>
</table>

New results in **bold**.
The remaining gap in **red**.