On the complexity of semantic self-minimization

AVoCS 2007

Adam Antonik &
Michael Huth
imperial.ac.uk/quads
Partial Kripke structures

- Often need aggressive abstraction of model prior to model checking
- Partial state spaces facilitate this, as Kripke structures with 3-valued labeling [Bruns & Godefroid 1999]
Abstraction-based model checking

- Partial Kripke structures have abstraction & refinement notion
- System = Kripke structure
- Abstraction = Partial Kripke structure, refined by System
- Verification Problem: “Do all Kripke structure refinements of Abstraction satisfy formula of $\mu$-calculus?”
  - If so, System will satisfy it, too.
  - If not, we may be no wiser.
Complexity, Soundness & Incompleteness

• Verification Problem: “Do all Kripke structure refinements of Abstraction satisfy formula of mu-calculus?”

• This is EXPTIME-complete in formula, quadratic in model [Bruns & Godefroid 2000]

• Approximate version of Verification Problem linear in formula/model [Bruns & Godefroid 1999]

• If approximate version verifies abstraction, system also verified (soundness)

• If approximate version doesn’t verify abstraction, system may still satisfy considered formula: under-approximation (incompleteness)
Refinement = Abstraction\(^{-1}\)

A binary relation \(\leq \subseteq S_M \times S_N\) is a refinement iff \(s \leq t\) implies

(a) \(L(s, q) \leq_i L(t, q)\) for all \(q \in \mathbb{AP}\),
(b) for all \((s, s') \in R_M\) there is \((t, t') \in R_N\) with \(s' \leq t'\), and
(c) for all \((t, t') \in R_N\) there is \((s, s') \in R_M\) with \(s' \leq t'\).

\[
\frac{1}{2} \leq_i 0 \quad \text{and} \quad \frac{1}{2} \leq_i 1
\]
Example

Pointed Kripke structure \((N, t1)\) refines pointed model \((M, s1)\)

\[ \preceq = \{(s_1, t_1), (s_2, t_2), (s_3, t_3), (s_4, t_3)\} \]
Formal Verification Problem

$(M,s)$ pointed model: $M$ with initial state $s$

\[ \forall \text{VAL}(M,s,\phi) \]

holds iff all pointed Kripke structures that refine $(M,s)$ satisfy $\phi$

\[ \exists \text{SAT}(M,s,\phi) \]

holds iff some pointed Kripke structure refines $(M,s)$ and satisfies $\phi$
Example

\[
\text{VAL}(M, s_1, \text{AF } (q \land \neg r)) \\
\text{VAL}(M, s_1, \text{AF EG } \neg r)
\]

Both judgments hold
Counterexample

\[ \text{VAL}(M, s_1, \text{AF AG } \neg r) \]

Doesn’t hold: \((N,t1)\) is counterexample
Approximate versions of judgments

- Use semantics similar to labeling algorithm
- Compositionally evaluate sub-formulas
- Do this in pessimistic and in optimistic mode for $\phi ::= q \mid Z \mid \phi \land \phi \mid \neg \phi \mid \text{EX} \phi \mid \mu Z. \phi$
- Pessimistic mode: under-approximates $\text{VAL}(M, s, \phi)$
- Optimistic mode: over-approximates $\text{SAT}(M, s, \phi)$
Optimistic (o) and pessimistic (p) approximative semantics for mu-calculus

Partial Kripke structure $M = (S, R, L)$

\[
\begin{align*}
\models q \models_o^\rho &= \{ s \mid L(s, q) \neq 0 \} \\
\models Z \models_o^\rho &= \rho(Z) \\
\models \phi \land \psi \models_o^\rho &= \models \phi \models_o^\rho \land \models \psi \models_o^\rho \\
\models \neg \phi \models_o^\rho &= S \setminus \models \phi \models_p^\rho \\
\models \text{EX} \phi \models_o^\rho &= \text{pre}(\models \phi \models_o^\rho) \\
\models \mu Z. \phi \models_o^\rho &= \text{lfp}\ F_{\phi, \rho}^o \\
\models q \models_p^\rho &= \{ s \mid L(s, q) = 1 \} \\
\models Z \models_p^\rho &= \rho(Z) \\
\models \phi \land \psi \models_p^\rho &= \models \phi \models_p^\rho \land \models \psi \models_p^\rho \\
\models \neg \phi \models_p^\rho &= S \setminus \models \phi \models_o^\rho \\
\models \text{EX} \phi \models_p^\rho &= \text{pre}(\models \phi \models_p^\rho) \\
\models \mu Z. \phi \models_p^\rho &= \text{lfp}\ F_{\phi, \rho}^p
\end{align*}
\]
Formal soundness of approximation

For sentence $\phi$ and for $m \in \{o, p\}$ set

$$(M, s) \models^m \phi \overset{\text{def}}{=} s \in \| \phi \|^m_\rho \text{ for some } \rho$$

Soundness as two, co-dependent, implications:

$$(M, s) \models^p \phi \text{ implies } \text{VALID}(M, s, \phi)$$

$$(M, s, \phi) \text{ implies } (M, s) \models^o \phi$$
Incompleteness of approximation

• If $L(s,q) = 1/2$, then the tautology $q \lor \neg q$ holds at state $s$, but the pessimistic semantics won’t verify this.

• But pessimistic semantics is complete for many practically relevant property patterns [Antonik & Huth 2006], e.g.

$$\neg \mathcal{E}[\neg q \mathcal{U} r] \land \mathcal{E}[\neg r \mathcal{U} (q \land \neg r \land \mathcal{E}[\neg s \mathcal{U} (r \land \neg s)])]$$

“precedence chain: 2 stimuli, 1 response; globally, $q$ and $s$ precede $r$”

patterns.project.cis.ksu.edu
Semantic self-minimization

• Sentence $\phi$ **pessimistically self-minimizing**: Iff for all pointed models $(M,s)$

\[
(M, s) \models^p \phi \iff \text{VAL}(M, s, \phi)
\]

• Sentence $\phi$ **optimistically self-minimizing**: Iff for all pointed models $(M,s)$

\[
(M, s) \models^o \phi \iff \text{SAT}(M, s, \phi)
\]
Decision Problems

- OSM = set of optimistically self-minimizing sentences
- PSM = set of pessimistically self-minimizing sentences
- VAL = set of valid sentences
- UNSAT = set of unsatisfiable sentences
Logics

Decision problems considered for three logics (i.e. for their sets of sentences):
• Propositional modal mu-calculus
• Propositional modal logic (no fixed points)
• Propositional logic (no fixed points, no modal operators)

Use generic variable for logics when convenient:

\[ L \in \{ \text{MC, PML, PL} \} \]
Partition

- Logic partitioned into six sets:

  I  VAL
  II UNSAT
  III OSM \ (VAL \cup PSM)
  IV PSM \ (UNSAT \cup OSM)
  V \lang \ (PSM \cup OSM)
  VI PSM \cap OSM
Partition in a picture

Negation maps pairs of sets into each other:
- OSM and PSM
- I and II
- III and IV
- V and itself
- VI and itself

Also, VAL in OSM and disjoint from PSM.
Dually, UNSAT in PSM and disjoint from OSM.
Sets I and II

Deciding membership in sets I and II has - of course - same complexity as deciding validity of logic:

- EXPTIME-complete for mu-calculus
- PSPACE-complete for modal logic
- coNP-complete for propositional logic
Set OSM, hardness result

- Sentence in OSM iff its negation is in PSM
- So OSM and PSM have same complexity
- Deciding OSM at least as hard as deciding validity of logic:

\[ E(\phi) = \phi \lor (x \land \neg x) \]

where x atomic proposition not occurring in \( \phi \)
- This is desired reduction to VAL:

\[ \phi \text{ is valid iff } E(\phi) \text{ is in OSM} \]
Hardness proof (for illustration)

• Let $\phi$ be valid, so $E(\phi)$ valid and so in OSM
• Let $E(\phi)$ be is OSM. Proof by contradiction: $\phi$ not valid, so $\phi$ false at some pointed Kripke structure $(K,t)$. Extend labeling $L$ of $K$ with $L(s,x) = 1/2$ for all states $s$, so $K$ now partial Kripke structure. We get

$$(K, t) \models^o E(\phi)$$

for this extended $K$, but no refining Kripke structure of extended $(K,t)$ satisfies $E(\phi)$ as this is semantically equivalent to $\phi$ on Kripke structures
Set OSM, upper bound for mu-calculus

- For mu-calculus, OSM in 2EXPTIME
- From $\phi$ construct two alternating tree automata - exponential blowup in worst case - and then do language inclusion check for these automata - exponential in size of these automata [Godefroid & Huth 2005]:
  \[ \mathcal{L}(A^3_{\models o}) \subseteq \mathcal{L}(A^3_{\phi}) \]
- This language inclusion checks "completeness half" of
  \[ (M, s) \models^0 \phi \iff SAT(M, s, \phi) \]
for all pointed models (M,s)
Set OSM, upper bound for modal logic

- For modal logic, OSM in EXPSPACE
- From $\phi$ construct two alternating tree automata as before, and again check

$$\mathcal{L}(A^3_\models_o) \subseteq \mathcal{L}(A^3_\phi)$$

- Both automata cannot distinguish trees at depths greater than size of $\phi$ (so called "shallow model property" of modal logic)
- So the above check is in PSPACE in the size of the automata
Set OSM, exact bound for propositional logic

- Already showed OSM is coNP-hard
- Show PL - OSM in NP:

```java
boolean NotInOSM(phi) {
    // **choose** model M such that M(x) = 1/2 for some x in AP(\phi);
    if (M |=^o \phi) {
        for (all x in AP(\phi) with M(x) = 1/2) {
            if (!(M[x -> 0] |=^o \phi) && !(M[x -> 1] |=^o \phi)) {
                ACCEPT;
            }
        }
        REJECT;
    }
}
```
Set III

Deciding set III at least as hard as deciding OSM, and so at least as hard as deciding VAL:

\[ F(\phi) = (\phi \lor x) \land (y \land (z \lor \neg z)) \]

where \( x, y, z \) atomic propositions not occurring in \( \phi \)

F is desired reduction:

\[ F(\phi) \in \text{III} \iff \phi \in \text{OSM} \]
Set III, upper bound

• Mu-calculus: set III in 2EXPTIME
• Modal logic: set III in EXPSPACE
• Propositional logic: set III in DP (as intersection of language in NP with language in coNP) and coNP-hard
  Don’t know whether set III is DP-complete.
Set V

Deciding set $V$ at least as hard as deciding **satisfiability** of the logic:

$$G(\phi) = (\phi \land (x \lor \neg x) \land y) \lor (z \land \neg z)$$

where $x, y, z$ atomic propositions not occurring in $\phi$

$G$ is desired reduction:

$$\phi \text{ satisfiable} \iff G(\phi) \in V$$

As upper bounds, we again get $2\text{EXPTIME}$, $\text{EXPSPACE}$, and $\text{NP(-complete)}$ - respectively
Set VI

- Formulas in set VI complete for optimistic and pessimistic approximate semantics.
- From our results we immediately get upper bounds $2\text{EPXTIME}$, $\text{EXPSPACE}$ and $\text{coNP}$ for mu-calculus, modal logic, and propositional logic (respectively):

$$(\phi \in PSM \cap OSM) \iff (\phi \in OSM \& \neg \phi \in OSM)$$
“Experimental” data

- Used Perl script to randomly generate “all” formulas of propositional logic for sizes 1 to 5
- Size = number of occurrences of logical connectives in formula
- Brute-force decision of membership: in OSM (~75%), in set VI (~50%), and in NP-complete set V (~2.45%)
- Less formulas seem to be in set VI as number of logical operators in formula increases
Summary of results for mu-calculus

<table>
<thead>
<tr>
<th>2EXPTIME, EXPTIME-hard</th>
<th>EXPTIME-complete</th>
<th>2EXPTIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSM</td>
<td>VAL</td>
<td>PSM ∩ OSM</td>
</tr>
<tr>
<td>PSM</td>
<td>UNSAT</td>
<td></td>
</tr>
<tr>
<td>OSM \ (VAL ∪ PSM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSM \ (UNSAT ∪ OSM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>μC \ (PSM ∪ OSM)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Summary of results for modal logic

<table>
<thead>
<tr>
<th>EXPSPACE, PSPACE-hard</th>
<th>PSPACE-complete</th>
<th>EXPSPACE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSM</td>
<td>VAL</td>
<td>PSM ∩ OSM</td>
</tr>
<tr>
<td>PSM</td>
<td>UNSAT</td>
<td></td>
</tr>
<tr>
<td>OSM \ (VAL ∪ PSM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSM \ (UNSAT ∪ OSM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PML \ (PSM ∪ OSM)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary of results for propositional logic

<table>
<thead>
<tr>
<th>DP, coNP-hard</th>
<th>NP-complete</th>
<th>coNP-complete</th>
<th>coNP</th>
</tr>
</thead>
<tbody>
<tr>
<td>( OSM \setminus (VAL \cup PSM) )</td>
<td>( \mathcal{P} \setminus (PSM \cup OSM) )</td>
<td>VAL</td>
<td>PSM \cap OSM</td>
</tr>
<tr>
<td>PSM \setminus (UNSAT \cup OSM)</td>
<td></td>
<td>UNSAT</td>
<td>OSM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PSM</td>
</tr>
</tbody>
</table>
Conclusions

• Studied complexity of deciding whether a formula loses precision in an approximate semantics for 3-valued models.
• For mu-calculus and modal logic, we showed that the complexity of validity is a lower bound for this, our upper bounds show an exponential gap.
• For propositional logic, this gap disappears.
• For propositional logic, deciding whether a formula loses precision for optimistic and pessimistic approximation matches complexity of satisfiability.
Future work

- Narrow currently exponential complexity gap for OSM, for mu-calculus and modal logic.
- Attempt hardness proofs for sets VI. Study Turing reductions.
- Study precision of patterns outside of LTL&CTL and ACTL.
- Study self-minimization for linear-time temporal logics.
Related work (not already discussed)

- [Larsen & Thomsen 1989] studied partial models of labeled transition systems and their refinement
- [Van Frassen 1966] defined and studied supervaluational meaning as precise version of approximate 3-valued semantics
- [Blamey 1980] proved that propositional logic formulas always have (in our terminology) semantically equivalent formulas that are self-minimizing
- [Reps et al. 2002] used BBDs and prime-implicants to implement semantic minimizations of propositional logic more efficiently

References to all related work are in the paper.
Thank you.
100 years of living science