The Complexity of Hybrid Logics

Thomas Schneider

School of Computer Science, University of Manchester, UK

Part of this work has been done jointly with Arne Meier, Martin Mundhenk, Michael Thomas, Volker Weber and Felix Weiß

18 June 2010
Overview

Starting point

expressive  convenient  well-behaved

Hybrid logics

E\downarrow x.\lozenge \downarrow y.\@_x\lozenge \neg y

often undecidable 😞

Question

**Question**  Under which restrictions can we get decidability back?

**Answer**  Allow only certain classes of frames!
Restrict combinations of operators! 😊
And now . . .

1. Hybrid Logic

2. $\mathcal{HL}$ over restricted frame classes

3. $\mathcal{HL}$ with restricted Boolean operators

4. Outlook
What is hybrid logic?

“Definition”

Hybrid logic = prop. logic + $\diamond \square$ + nominals + $@\downarrow\exists\forall E A \ldots$

moder logic

HL speaks about frames and models.
What is hybrid logic?

"Definition"

Hybrid logic = prop. logic + ◊ □ + nominals + @↓ ∀ E A \ldots
modal logic

◊ \varphi \quad \text{in some successor, } \varphi

□ \varphi \quad \text{in all successors, } \varphi
What is hybrid logic?

"Definition"

Hybrid logic = prop. logic + ◇□ + nominals + @↓∃∀EA ... modal logic

\( i \)  name for a state
\( @i \varphi \)  at state named \( i \), \( \varphi \)
What is hybrid logic?

“Definition”

Hybrid logic = prop. logic + modal logic + nominals + @↓∃∀EA . . .

\[ \downarrow x.\varphi \text{ with } x \text{ bound to } \text{current state}, \varphi \]
\[ \exists x.\varphi \text{ with } x \text{ bound to } \text{some state}, \varphi \]
\[ \forall x.\varphi \text{ with } x \text{ bound to } \text{any state}, \varphi \]
What is hybrid logic?

“Definition”

Hybrid logic = prop. logic + \( \Diamond \Box \) + nominals + \( @\downarrow \exists \forall \text{EA} \ldots \)

\[ E \varphi \quad \text{in some state, } \varphi \]
\[ A \varphi \quad \text{in all states, } \varphi \]
Hybrid temporal logic

“Definition”

Hybrid temporal logic = hybrid logic − ◊□ + FG PH US ...
Hybrid temporal logic

“Definition”

Hybrid temporal logic = hybrid logic − ◻◻ + FG PH US ...

Until

ϕUψ in some successor, ψ, and from here until there, ϕ

Since

S = U⁻¹
A bit of notation

- Relevant operators: $\text{F P U S @ } \downarrow \exists E$
  (These are just duals: $\text{G H } \forall A$)

- Consider languages containing $\text{F}$
  and arbitrary combinations of $\text{P U S @ } \downarrow \exists E$
A bit of notation

- Relevant operators: $F \ P \ U \ S \ @ \ \downarrow \ \exists \ \forall \ E$
  (These are just duals: $G \ H \ A$)

- Consider languages containing $F$
  and arbitrary combinations of $P \ U \ S \ @ \ \downarrow \ \exists \ E$

- Write languages as follows
  \[ \mathcal{ML}(F) \] basic modal language $K$
  \[ \mathcal{HL}(F, @) \] basic hybrid language
  \[ \mathcal{HL}(F, \downarrow, E) \] a very expressive hybrid language
Decision problems

**Satisfiability problem** \( \mathcal{HL}(\cdot)\text{-SAT} \)

**Input** \( \varphi \in \mathcal{HL}(\cdot) \)

**Question** Are there \( \mathcal{M}, g, m \in \mathcal{M} \) such that \( \mathcal{M}, g, m \models \varphi \)?

**Model-checking problem** \( \mathcal{HL}(\cdot)\text{-MC} \)

**Input** \( \varphi \in \mathcal{HL}(\cdot), \mathcal{M}, g \)

**Question** Is there \( m \in \mathcal{M} \) such that \( \mathcal{M}, g, m \models \varphi \)?

We will focus on SAT here.
# Complexity classes

<table>
<thead>
<tr>
<th>Name</th>
<th>Meaning</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>L</strong></td>
<td>logarithmic space</td>
<td>graph accessibility</td>
</tr>
<tr>
<td><strong>P</strong></td>
<td>polynomial time</td>
<td>(model checking)</td>
</tr>
<tr>
<td><strong>NP</strong></td>
<td>nondeterministic pol. time</td>
<td>prop. logic SAT</td>
</tr>
<tr>
<td><strong>PSPACE</strong></td>
<td>polynomial space</td>
<td>modal logic SAT</td>
</tr>
<tr>
<td><strong>EXP</strong></td>
<td>exponential time</td>
<td></td>
</tr>
<tr>
<td><strong>NEXP</strong></td>
<td>nondeterministic exp. time</td>
<td></td>
</tr>
<tr>
<td><strong>N2EXP</strong></td>
<td>nondeterministic 2× exp. time</td>
<td></td>
</tr>
<tr>
<td><strong>n.d.</strong></td>
<td>nonelementarily decidable</td>
<td></td>
</tr>
<tr>
<td><strong>coRE</strong></td>
<td>undecidable</td>
<td></td>
</tr>
</tbody>
</table>

- **EXP**: exponential time
- **NEXP**: nondeterministic exponential time
- **N2EXP**: nondeterministic 2× exponential time
- **n.d.**: nonelementarily decidable
- **coRE**: undecidable

- **HL SAT**: Hybrid Logic SAT
- **FOL SAT**: First-Order Logic SAT
"Traditional" complexity results for SAT

<table>
<thead>
<tr>
<th>Language</th>
<th>Completeness</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{ML}(F)$</td>
<td>PSPACE</td>
<td>Ladner 77</td>
</tr>
<tr>
<td>$\mathcal{ML}(F, P)$</td>
<td>PSPACE</td>
<td>Spaan 93</td>
</tr>
<tr>
<td>$\mathcal{HL}(F, @)$</td>
<td>PSPACE</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>$\mathcal{HL}(F, P)$</td>
<td>EXP</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>$\mathcal{HL}(F, P, @)$</td>
<td>EXP</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>$\mathcal{HL}(U, S, E)$</td>
<td>EXP</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>$\mathcal{HL}(F, \downarrow)$</td>
<td>coRE</td>
<td>Blackburn et al. 95, Goranko 96, Areces et al. 99</td>
</tr>
</tbody>
</table>
### “Traditional” complexity results for SAT

<table>
<thead>
<tr>
<th>Language</th>
<th>Completeness</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mathcal{ML}(F) )</td>
<td>PSPACE</td>
<td>Ladner 77</td>
</tr>
<tr>
<td>( \mathcal{ML}(F, P) )</td>
<td>PSPACE</td>
<td>Spaan 93</td>
</tr>
<tr>
<td>( \mathcal{HL}(F, @) )</td>
<td>PSPACE</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>( \mathcal{HL}(F, P) )</td>
<td>EXP</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>( \mathcal{HL}(F, P, @) )</td>
<td>EXP</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>( \mathcal{HL}(U, S, E) )</td>
<td>EXP</td>
<td>Areces et al. 99</td>
</tr>
<tr>
<td>( \mathcal{HL}(F, \downarrow) )</td>
<td>coRE</td>
<td>Blackburn et al. 95, Goranko 96, Areces et al. 99</td>
</tr>
</tbody>
</table>

How can we tame \( \downarrow \)?
And now ...

1. Hybrid Logic

2. $\mathcal{HL}$ over restricted frame classes

3. $\mathcal{HL}$ with restricted Boolean operators

4. Outlook
Applications often require frames with certain properties.

**Example: temporal logic**

<table>
<thead>
<tr>
<th>States</th>
<th>$mRm'$</th>
<th>$\lozenge \varphi$ (F$\varphi$)</th>
<th>$\square \varphi$ (G$\varphi$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\triangleq$ points in time</td>
<td>$\triangleq$ “$m'$ is in the future of $m$”</td>
<td>$\triangleq$ “at some time in the future, $\varphi$”</td>
<td>$\triangleq$ “always in the future, $\varphi$”</td>
</tr>
</tbody>
</table>

**Relevant classes of frames:**

- linear orders
- transitive trees
Applications often require frames with certain properties.

**Example: epistemic logic**

- **States** $\cong$ possible worlds of an agent
- $mRm'$ $\cong$ “being in world $m$, the agent thinks $m'$ possible”
- $\Diamond \varphi (\hat{K}\varphi) \cong$ “the agent considers $\varphi$ possible”
- $\Box \varphi (K\varphi) \cong$ “the agent knows that $\varphi$”

Relevant classes of frames:

- frames with equivalence relations
- superclasses thereof, e.g., transitive frames
SAT over restricted frames

Satisfiability problem \( \mathcal{HL}(\cdot)-\mathfrak{F}\text{-SAT} \)

**Input** \( \varphi \in \mathcal{HL}(\cdot) \)

**Question** Are there \( \mathcal{M} \in \mathfrak{F}, g, m \in \mathcal{M} \) with \( \mathcal{M}, g, m \models \varphi \)?
SAT over restricted frames

**Satisfiability problem** $\mathcal{HL}(\cdot)$-SAT

**Input** $\varphi \in \mathcal{HL}(\cdot)$

**Question** Are there $M \in F$, $g$, $m \in M$ with $M, g, m \models \varphi$?

<table>
<thead>
<tr>
<th>Language</th>
<th>Frame class</th>
<th>Completeness</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{ML}(F)$</td>
<td>equiv</td>
<td>NP</td>
<td>Ladner 77</td>
</tr>
<tr>
<td>$\mathcal{ML}(F, P)$</td>
<td>lin</td>
<td>NP</td>
<td>Ono, Nakamura 80</td>
</tr>
<tr>
<td>$\mathcal{HL}(F, P, E)$</td>
<td>lin</td>
<td>NP</td>
<td>Areces et al. 00</td>
</tr>
<tr>
<td>$\mathcal{HL}(U, S, E)$</td>
<td>$(\mathbb{N}, \prec)$</td>
<td>PSPACE</td>
<td>Areces et al. 00</td>
</tr>
<tr>
<td>$\mathcal{HL}(F, \downarrow, E)$</td>
<td>lin</td>
<td>n.d.</td>
<td>Francesch et al. 03</td>
</tr>
<tr>
<td>$\mathcal{HL}(F, \downarrow)$</td>
<td>trans, equiv</td>
<td>NEXP</td>
<td>Mundhenk et al. 05</td>
</tr>
</tbody>
</table>
A more systematic approach

Examine complexity of SAT for all hybrid languages with $F$ and arbitrary combinations of $P \cup S \downarrow \exists E$ over

- all frames
- transitive frames
- transitive trees
- linear orders
- $(\mathbb{N}, <)$
- frames with equivalence relations
The lattice of languages
Complexity results over arbitrary frames

PSPACE

EXP

core
Complexity results over transitive frames

- **PSPACE**: in 2EXP, EXP-hard
- **EXP**: in EXP-hard
- **NEXP**: in coRE, NEXP-hard
- **coRE**: in coRE

### Hybrid Logic
- Restricted frame classes
- Restricted Boolean operators

### Outlook
Complexity results over transitive trees
Complexity results over linear orders

nonelementarily decidable

decidable, PSPACE-hard
Complexity results over \((\mathbb{N}, <)\)
Complexity results over equivalence relations

Diagram showing the relationships between different complexity classes:

- NP
- NEXP
- N2EXP

The diagram illustrates the hierarchy and relationships among these classes, with specific operators and restrictions denoted by symbols such as ♦ and @, indicating different logical or relational operations.
For these six frame classes, $\mathcal{H}\mathcal{L}(\Diamond_1, \Diamond_2, \downarrow)$-$\mathcal{F}$-SAT is already coRE-complete.
And now ...

1. Hybrid Logic
2. \( \mathcal{HL} \) over restricted frame classes
3. \( \mathcal{HL} \) with restricted Boolean operators
4. Outlook
Propositional fragments of $\mathcal{HL}$

Restrict the set of *propositional* operators!

- Why?
  
  Propositional SAT becomes tractable, e.g., without negation.  
  (Lewis ’79)

  SAT for $\mathcal{ML}$ or LTL becomes tractable for certain restrictions.  
  (Bauland et al. ’06/07)

  SAT for many sub-Boolean description logics is tractable.  
  (Baader et al. ’98/05/08, Calvanese et al. ’05–07)

- 3 parameters:

  - frame class $F$
  - set $O$ of modal/hybrid operators
  - set $B$ of Boolean operators

  $\Rightarrow \quad \mathcal{HL}(O, B)$-$\exists$-SAT
Classify $\mathcal{HL}(O, B)$-$\exists\forall$-SAT for decidability and complexity w.r.t.
- all $B$
- $O$ with $\{\Diamond, \downarrow\} \subseteq O \subseteq \{\Diamond, \Box, \downarrow, @\}$
- $F \in \{\text{all, trans, equiv, serial}\}$

- Find border between decidable and undecidable fragments
- Find tight complexity bounds
**Complexity results over arbitrary frames**

<table>
<thead>
<tr>
<th><strong>$\mathcal{HL}(O, B)$-all-SAT</strong></th>
<th><strong>is . . .</strong></th>
</tr>
</thead>
</table>
| **coRE-compl.**                          | if $B$ can express $x \land \neg y$  
|                                          | or all self-dual functions  |
| **coNP-hard**                            | if $B$ contains $\land$ and $\Box \in O$  |
| **in L**                                 | if $B$ can express only $\land, \lor, \top, \bot$ and $\Box \notin O$  
|                                          | or $B$ can express only $\lor, \top, \bot$ or only $\neg, \top, \bot$  |
| **trivial**                              | in almost all other cases  |
### Complexity results over arbitrary frames

<table>
<thead>
<tr>
<th>Classification</th>
<th>Condition</th>
</tr>
</thead>
</table>
| **coRE**-compl. | if $B$ can express $x \wedge \neg y$  
                            or all self-dual functions                                               |
| **coNP**-hard   | if $B$ contains $\land$ and $\Box \in O$                                  |
| in **L**        | if $B$ can express only $\land, \lor, T, \bot$ and $\Box \notin O$       
                            or $B$ can express only $\lor, T, \bot$ or only $\neg, T, \bot$        |
| trivial         | in almost all other cases                                                  |

Almost the same classification for $\mathcal{HL}(O, B)$-trans-SAT
Complexity results over serial frames

\[ \mathcal{HL}(O, B) \]-serial-SAT is . . .

\textbf{coRE-compl.} if \( B \) can express \( x \land \neg y \)

or all self-dual functions

\textbf{in L} if \( B \) can express only monotone functions

or \( B \) can express only \( \neg, T, \bot \)

\textbf{trivial} in almost all other cases
\( \mathcal{HL}(O, B) \)-equiv-SAT is ... 

- \textbf{NEXP}-compl. if \( B \) contains \( x \land \neg y \) or \( B \) all self-dual functions
- in \textbf{L} if \( B \) can express only monotone functions or \( B \) can express only \( \neg, T, \bot \)
- trivial in almost all other cases
And now . . .

1. Hybrid Logic
2. $\mathcal{HL}$ over restricted frame classes
3. $\mathcal{HL}$ with restricted Boolean operators
4. Outlook
Modularity of specifications

Specification $\varphi_1 \land \varphi_2$ refines $\varphi_1$ if:
for every $\psi$ that uses only symbols from $\varphi_1$:
if $\varphi_1 \land \varphi_2 \models \psi$, then $\varphi_1 \models \psi$.

If we’re only interested in the part of a theory that speaks about a certain subsignature, we can “forget” unnecessary conjuncts.
Modularity of specifications

Specification $\varphi_1 \land \varphi_2$ refines $\varphi_1$ if:
for every $\psi$ that uses only symbols from $\varphi_1$:
if $\varphi_1 \land \varphi_2 \models \psi$, then $\varphi_1 \models \psi$.

$\leadsto$ If we’re only interested in the part of a theory that speaks about a certain subsignature, we can “forget” unnecessary conjuncts.

We also say $\varphi_1 \land \varphi_2$ is a conservative extension of $\varphi_1$. 
Deciding and approximating conservativity

- Deciding conservativity is
  - at least as hard as satisfiability
  - coNEXP-complete for $\mathcal{ML}(\Diamond)$
  - undecidable for description logics (DLs) with nominals

- Sufficient conditions for conservativity in expressive DLs exist
  - efficient module extraction algorithms

[Ghilardi at al. 06]
[Ghilardi at al. 06]
[Lutz et al. 07]
Deciding and approximating conservativity

- Deciding conservativity is
  - at least as hard as satisfiability
  - coNEXP-complete for $\mathcal{ML}(\Diamond)$
  - undecidable for description logics (DLs) with nominals

- Sufficient conditions for conservativity in expressive DLs exist
  $\rightsquigarrow$ efficient module extraction algorithms

**Carry over insights to hybrid logics:**

- Devise module notions for HL similar to locality
- Find efficient algorithms for refinement test, module extraction
Deciding and approximating conservativity

- Deciding conservativity is
  - at least as hard as satisfiability
  - \textit{coNEXP}-complete for $\mathcal{ML}(\Diamond)$
  - undecidable for description logics (DLs) with nominals

- Sufficient conditions for conservativity in expressive DLs exist
  $\leadsto$ efficient module extraction algorithms

---

Carry over insights to hybrid logics:

- Devise module notions for HL similar to locality
- Find efficient algorithms for refinement test, module extraction

Thank you.