Game Logic
— CoalgebraicCompleteness and Automata
Helle Hvid Hansen, TU Delft

ALCOP, University of Strathclyde, 10-12 April 2017
Introduction

**Modal logics:**
- Versatile logics for reasoning about state-based systems.
- Good trade-off between expressiveness and decidability.
- Established as the logics of coalgebras.

**Aim:** Coalgebraic understanding of dynamic modal logics, like Propositional Dynamic Logic (PDL) and Game Logic.
- Identify relevant mathematical structure.
- Framework for developing dynamic coalgebraic logics.
- Transfer insights to other dynamic/game settings.
- Improve our understanding of fragments of fixpoint logics.
Game Logic (GL)

Rohit Parikh, “The Logic of Games and its Applications”.

- Strategic ability in determined 2-player games.
  
  $\langle \alpha \rangle \varphi$ expresses

  “player I has strategy in $\alpha$ to ensure outcome satisfies $\varphi$”

- Game version of PDL:
  - PDL: 1-player game (nondeterministic programs)
  - from program constructs to game constructs.
Determined 2-Player Games

Typical examples: 2-player, extensive games with perfect information.
Example: Players: I (black) and II (white), moves: L or R.

\[
\begin{array}{c}
\text{I} \\
L \\
R \\
\text{II} \\
\text{I}
\end{array}
\]

Strategic ability | formula | strategy
---|---|---
I can ensure \( p \): & \( \langle \alpha \rangle p \) & LR
I cannot ensure \( q \): & \( \neg \langle \alpha \rangle q \)
I cannot ensure \( r \): & \( \neg \langle \alpha \rangle r \)
II can ensure \( q \): & \([\alpha]q\) & L
II can ensure \( \neg r \): & \([\alpha]r\) & L

Note that: \( \langle \alpha \rangle (q \lor r) \), but \( \neg \langle \alpha \rangle q \) and \( \neg \langle \alpha \rangle r \).

Game modalities are not disjunctive.
They are only monotonic: \( \langle \alpha \rangle p \rightarrow \langle \alpha \rangle (p \lor q) \)
Determined 2-Player Games

Typical examples: 2-player, extensive games with perfect information.

Example: Players: I (black) and II (white), moves: L or R.

Strategic normal form:

\[
\begin{array}{c|cc}
L & p & q \\
R & q & r \\
\end{array}
\]

\[
\begin{array}{c|cc}
L & p,q & r \\
R & p,q & p \\
\end{array}
\]

\[
\begin{array}{c|cc}
L & q & r \\
R & q & p \\
\end{array}
\]

Determinacy: \( \langle \alpha \rangle \varphi \leftrightarrow \neg [\alpha] \neg \varphi \)

(“I can ensure \( \varphi \) iff II cannot avoid \( \varphi \)”)

Helle Hvid Hansen (TU Delft) Game Logic, Completeness and Automata ALCOP 11/4/2017 7 / 54
Effectivity in State-Based Game Models

- Games are played in the context of a state space $X$.
- Game outcomes are associated with states.

Player I is effective for $U \subseteq X$ if I has a strategy to ensure the outcome is in $U$.

Player I is effective for:

- $\{x_1, x_3\}$
- $\{x_1, x_4\}$
- $\{x_2, x_3\}$
- $\{x_2, x_4\}$
- and all supersets of those.

Player II is effective for:

- $\{x_1, x_2\}$ and $\{x_3, x_4\}$
- and all supersets of those.
Game Structures

- Let $E_\alpha(x) \subseteq \mathcal{P}(X)$ be defined by:
  
  \[ U \in E_\alpha(x) \text{ iff player 1 is effective for } U \text{ in } \alpha \text{ starting in } x. \]

  Then: $U \in E_\alpha(x)$ and $U \subseteq U' \implies U' \in E_\alpha(x)$. 

- Let $\mathcal{M}$ be the monotone neighbourhood functor:
  
  \[ \mathcal{M}(X) = \{ N \subseteq \mathcal{P}(X) \mid U \in N, U \subseteq U' \implies U' \in N \} \]
  
  $\mathcal{M}(f) = (f^{-1})^{-1}$

- A game frame is
  
  - a multi-modal monotonic neighbourhood frame
    
    \[ F = (X, \{ E_\alpha : X \to \mathcal{M}(X) \mid \alpha \in A \}) \]
  
  - a coalgebra $F : X \to (\mathcal{M}(X))^A$

- A game model $M = (F, V)$ is a game frame $F$ with a valuation $V : X \to \mathcal{P}(P_0)$. 

Game Logic Syntax

formulas $\varphi ::= p \in P_0 \mid \neg \varphi \mid \varphi \land \varphi \mid \langle \alpha \rangle \varphi$

games $\alpha ::= a \in A_0 \mid \alpha; \alpha \mid \alpha \cup \alpha \mid \alpha^* \mid \varphi? \mid \alpha^d$

Game operations:

- (composition) $\alpha_1; \alpha_2$: play $\alpha_1$ then $\alpha_2$,
- (angelic choice) $\alpha_1 \cup \alpha_2$: player 1 chooses between $\alpha_1$ or $\alpha_2$,
- (angelic iteration) $\alpha^*$: $\alpha$ is played repeatedly (possibly 0 times), after each round, player I chooses whether to continue.
- (dual) $\alpha^d$: players switch roles in $\alpha$.
- (tests) $\varphi?$: if $\varphi$ holds then continue, otherwise player I loses.
Standard Game Models

(similar to standard PDL model)

- (composition)
  \[ U \in E_{\alpha_1;\alpha_2}(x) \text{ iff } \exists V \in E_{\alpha_1}(x) : \forall v \in V : X \in E_{\alpha_2}(v). \]

- (angelic choice) \( E_{\alpha_1 \cup \alpha_2}(x) = E_{\alpha_1}(x) \cup E_{\alpha_2}(x) \)

- (angelic iteration)
  \[ U \in E_{\alpha^*}(x) \text{ iff } x \in \hat{E}_{\alpha^*}(U) \text{ where } \hat{E}_{\alpha^*}(U) = \mu X. U \cup \hat{E}_{\alpha}(X). \]

(after each round, player I chooses whether to continue).

- (dual)
  \[ U \in E_{\alpha^d}(x) \text{ iff } X \setminus U \notin E_{\alpha}(x). \]
Axiomatisation and Completeness

- **GL = monotonic modal logic M** (ML of mon. nbhd. frames) plus

  \[ \langle \alpha; \delta \rangle \varphi \leftrightarrow \langle \alpha \rangle \langle \delta \rangle \varphi \]
  \[ \langle \psi ? \rangle \varphi \leftrightarrow (\psi \land \varphi) \]
  \[ \varphi \lor \langle \alpha \rangle \langle \alpha^* \rangle \varphi \rightarrow \langle \alpha^* \rangle \varphi \]
  \[ \langle \alpha \cup \delta \rangle \varphi \leftrightarrow \langle \alpha \rangle \varphi \lor \langle \delta \rangle \varphi \]
  \[ \langle \alpha^d \rangle \varphi \leftrightarrow \neg \langle \alpha \rangle \neg \varphi \]
  \[ \varphi \lor \langle \alpha \rangle \varphi \rightarrow \psi \]
  \[ \langle \alpha^* \rangle \varphi \rightarrow \psi \]

- **Without dual:** sound and complete [Parikh 1985].
- **Without iteration:** sound and strongly complete [Pauly 2001].
- Completeness of full GL still open question.
Coalgebraic Dynamic Logic

Joint work with Clemens Kupke
A General Picture...?

<table>
<thead>
<tr>
<th>PDL</th>
<th>GL</th>
<th>Basic set up</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Kripke semantics</strong></td>
<td><strong>Mon. nbhd. semantics</strong></td>
<td><strong>T-Coalgebra semantics</strong></td>
</tr>
<tr>
<td>$X \xrightarrow{\alpha} P^X$</td>
<td>$X \xrightarrow{\alpha} M^X$</td>
<td>$X \xrightarrow{\alpha} T^X$</td>
</tr>
<tr>
<td>$X \rightarrow (P^X)^A$</td>
<td>$X \rightarrow (M^X)^A$</td>
<td>$X \rightarrow (T^X)^A$</td>
</tr>
<tr>
<td>$; , \cup , (\neg)^* , ?$</td>
<td>$; , \cup , (\neg)^* , ? , (\neg)^d$</td>
<td>$T$-coalg. operations (?)</td>
</tr>
<tr>
<td>Normal ML <strong>K</strong> plus reduction axioms</td>
<td>Monotonic ML <strong>M</strong> plus reduction axioms</td>
<td>$T$-Coalgebraic ML plus</td>
</tr>
<tr>
<td></td>
<td></td>
<td>reduction axioms (?)</td>
</tr>
<tr>
<td>$f : X \rightarrow Y$</td>
<td>$f : X \rightarrow Y$</td>
<td>$f : X \rightarrow Y$</td>
</tr>
<tr>
<td>$\mathcal{P}(f) = f[\emptyset]$</td>
<td>$\mathcal{M}(f) = (f^{-1})^{-1}$</td>
<td>$T(f) : T^X \rightarrow T^Y$</td>
</tr>
<tr>
<td>(direct image)</td>
<td>(double-inv. image)</td>
<td></td>
</tr>
</tbody>
</table>
Coalgebraic Modal Logic

Basic idea

Basic Modal Logic

\[ \text{Kripke frames } X \rightarrow \mathcal{P}(X) \]

= 

Coalgebraic Modal Logic

\[ \text{T-coalgebras } X \rightarrow T(X) \]

Develop modal logic for T-coalgebras, \textit{parametric} in \( T : C \rightarrow C \).

Syntax

Given a collection of modal operators \( \Lambda \) and a set \( P_0 \) of propositional variables. The set \( \mathcal{F}(\Lambda) \) of formulas over \( \Lambda \) is defined as follows:

\[ \mathcal{F}(\Lambda) \ni \varphi ::= p \in P_0 \mid \bot \mid \neg \varphi \mid \varphi \land \varphi \mid \Diamond \varphi, \quad \Diamond \in \Lambda \]

(We only consider unary modalities \( \Diamond \))
Coalgebraic Modal Logic: Semantics
cf. [Pattinson, Roessiger]

\( T \)-coalgebraic semantics consists of:

- a functor \( T : \text{Set} \to \text{Set} \)
- for every modal operator \( \Diamond \in \Lambda \), a natural transformation

\[
\Diamond : Q \Rightarrow QT \quad \text{(predicate lifting)}
\]

where \( Q \) denotes the contravariant power set functor
\( (QX = 2^X, Q(f) = f^{-1}) \), so \( \Diamond_X : 2^X \to 2^{TX} \).

Truth in \( T \)-model \((X, \gamma : X \to TX, V : P_0 \to \mathcal{P}X)\)

\[
\begin{align*}
[p] &= V(p) \quad \text{for } p \in P_0 \\
\vdots \\
[\Diamond \varphi] &= \gamma^{-1}(\Diamond_X([\varphi]))
\end{align*}
\]
Equivalently...

There is a well-known one-to-one correspondence between:

- \( \diamondsuit : Q \Rightarrow QT \) \((\diamondsuit_X : 2^X \rightarrow 2^{TX})\)
- \( \hat{\diamondsuit} : T \Rightarrow Q^{\text{op}}Q \) \((\text{transpose}\ \hat{\diamondsuit}_X : TX \rightarrow 2^{2^X})\)
- \( \check{\diamondsuit} : T2 \rightarrow 2 \) \((\text{Yoneda})\) \("\text{allowed 0-1 patterns}\)"

Examples:

- Kripke box: \( \diamondsuit_X(U) = \{ V \subseteq X \mid V \subseteq U \} \), \( \hat{\diamondsuit}_X(V) = \{ U \subseteq X \mid V \subseteq U \} \) and \( \check{\diamondsuit}(V \in \mathcal{P}2) = 1 \) iff \( 0 \notin V \)

- Mon. nbhd. diamond: \( \diamondsuit_X(U) = \{ N \in \mathcal{M}X \mid U \in N \} \), \( \hat{\diamondsuit}_X(N) = N \), \( \check{\diamondsuit}(N \in \mathcal{M}2) = 1 \) iff \( \{1\} \in N \)
3.1 Dynamic Syntax and Semantics
Two perspectives:

\[ \xi : X \rightarrow (TX)^A \quad T^A\text{-coalgebra, modalities} \]

\[ \hat{\xi} : A \rightarrow (TX)^X \quad \text{algebra homomorphism, program operations} \]
Dynamic Syntax

Given

- $\Sigma$, a signature (functor).
- $P_0$, a countable set of atomic propositions.
- $A_0$, a countable set of atomic programs.

we define

formulas $\mathcal{F} \ni \varphi ::= p \in P_0 \mid \neg \varphi \mid \varphi \lor \varphi \mid \langle \alpha \rangle \varphi$

programs $A \ni \alpha ::= a \in A_0 \mid \alpha; \alpha \mid \sigma(\alpha_1, \ldots, \alpha_n) \mid \alpha^* \mid \varphi ?$

where $\sigma \in \Sigma$ is $n$-ary operation symbol.
3.2 Operations on $T$-coalgebras
Program Operations from Monads
(cf. Moggi, and many others)

- Monad $T$ encodes computational effects (non-determinism, exceptions, continuations, input/output,...)
- Kleisli arrows $X \to TY$ are functional programs.

- A monad is functor $T : \text{Set} \to \text{Set}$ together with natural transformations
  $\eta : \text{Id} \Rightarrow T(\text{unit})$ and $\mu : T \circ T \Rightarrow T$ (multiplication)
  satisfying certain axioms...
- Sequential composition is Kleisli composition.
  
  \[ ( X \xrightarrow{\alpha} TX ) \star_T ( X \xrightarrow{\beta} TX ) = 
  X \xrightarrow{\alpha} TX \xrightarrow{T\beta} TTX \xrightarrow{\mu_X} TX \]
- Skip program is unit: $\eta_X : X \to TX$. 
\( \mathcal{P} \) and \( \mathcal{M} \) are monads

- \( \mathcal{P} \) is monad \((\mathcal{P}, \eta, \mu)\) with:
  \[
  \eta_X(x) = \{x\},
  \]
  \[
  \mu_X(\{U_i \mid i \in I\}) = \bigcup_{i \in I} U_i.
  \]
  (*\(\mathcal{P}\) is relation composition, \(\eta_X = \text{Id}_X\))

- \( \mathcal{M} \) is a monad \((\mathcal{M}, \eta, \mu)\) with:
  \[
  \eta_X(x) = \{U \subseteq X \mid x \in U\}
  \]
  \[
  \mu_X(W) = \{U \subseteq X \mid \eta_\mathcal{P}(X)(U) \in W\}
  \]
  (*\(\mathcal{M}\) is composition of effectivity functions)

- \( \mathcal{N} = \mathcal{Q}^\text{op} \mathcal{Q} \) is a monad \((\eta \text{ and } \mu \text{ as for } \mathcal{M})\):
Dynamic Monads

Iteration requires extra structure.

A monad \((T, \mu, \eta)\) is called dynamic if

- For all sets \(X\), \(TX\) can be equipped with a sup-lattice structure \((TX, \bigvee)\) (i.e., a complete join semilattice).
  
  (We denote the empty join in \(TX\) by \(\bot_{TX}\).)

- Lift \(\bigvee\) pointwise to the Kleisli Hom-sets \(K\ell(T)(X, X)\), then Kleisli-composition is monotone:

  \[
  \forall f, g_1, g_2 : X \rightarrow TX : g_1 \leq g_2 \implies f \ast g_1 \leq f \ast g_2.
  \]

In FICS 2015 paper: we assumed that \(\ast\) left-distributes over join.

- Bad news: Doesn’t seem to hold for \(M\).
- Good news: We don’t need it! (Monotonicity suffices)
Iteration and Tests

Let \((T, \eta, \mu)\) be a dynamic monad.

**Iteration:** For a map \(\alpha : X \to TX\), we define \(\alpha^* = \text{LFP} \cdot \Phi_\alpha\) where

\[
\Phi_\alpha : \mathcal{K}\ell(T) \to \mathcal{K}\ell(T)
\]

\[
g \mapsto \eta_X \lor (\alpha \ast g)
\]

**Tests:** For a formula \(\varphi\), we define \(\alpha = \varphi?\) via Kleisli identity and empty join \(\bot = \bigvee \emptyset \in TX:\)

\[
\varphi?(x) = \eta_X(x) \quad \text{if} \quad x \in \llbracket \varphi \rrbracket^m, \quad \text{else} \bot.
\]
Pointwise Operations

- An \( n \)-ary natural operation on \( T \) is a natural transformation

\[
\sigma : T^n \Rightarrow T \quad (Tf \text{ preserves } \sigma)
\]

- \( \sigma : T^n \Rightarrow T \) yields pointwise operation on \((TX)^X\), e.g.,

\[
\sigma^X_X(c_1, c_2)(x) = \sigma_X(c_1(x), c_2(x))
\]

- Given finitary signature functor \( \Sigma \), a natural \( \Sigma \)-algebra is natural transformation \( \theta : \Sigma T \Rightarrow T \) and yields pointwise \( \Sigma \)-algebra on \((TX)^X\):

\[
\theta^X_X : \Sigma((TX)^X) \rightarrow (TX)^X
\]
Natural Pointwise Operations: Examples

Natural operations on $\mathcal{P}$:

- Union $\cup: \mathcal{P} \times \mathcal{P} \Rightarrow \mathcal{P}$ is natural operation, since

  $$f[U \cup U'] = f[U] \cup f[U'] \quad (\mathcal{P}f(U) = f[U])$$

The pointwise extension of $\cup: \mathcal{P} \times \mathcal{P} \Rightarrow \mathcal{P}$ is union of relations $(R_1 \cup R_2)(x) = R_1(x) \cup R_2(x)$.

- Note: Intersection, complement are not natural on $\mathcal{P}$.

Natural operations on $\mathcal{M}$:

- All Boolean operations (since preserved by $f^{-1}$).

- Dual operation $d: \mathcal{M} \Rightarrow \mathcal{M}$ where for all $N \in \mathcal{M}(X)$, and $U \subseteq X$, $U \in d_X(N)$ iff $X \setminus U \notin N$.

  Game operation $(-)^d$ is the pointwise extension.
Summary of Requirements for Coalgebraic Dynamic Semantics

We assume given:

- set $A_0$ of atomic programs.
- set $P_0$ of atomic propositions.

We require for dynamic $T$-coalgebra semantics:

- $\Diamond: Q \Rightarrow Q \circ T$ is predicate lifting (for modalities)
- $T = (T, \eta, \mu)$, a monad (for sequential comp.)
- $\mathcal{T} = (T, \eta, \mu)$, is a dynamic monad (for iteration and tests)
- $\theta: \Sigma T \Rightarrow T$, a natural $\Sigma$-algebra (for pointwise ops)
Standard Dynamic Models

Def. A $\theta$-dynamic $T$-model is a triple $\mathcal{M} = (X, \gamma: X \rightarrow (TX)^A, V)$ where

- $\widehat{\gamma}|_{A_0} = \widehat{\gamma}_0: A_0 \rightarrow (TX)^X$ interprets atomic programs,
- sequential composition, iteration, tests and pointwise operations are defined compositionally from $\gamma_0$ as described.
- $V: P_0 \rightarrow P(X)$ is a valuation of atomic propositions.
- Modalities are interpreted by $\diamondsuit$:

$$\llbracket\langle\alpha\rangle \varphi\rrbracket = \widehat{\gamma}(\alpha)^{-1}(\diamondsuit_X(\llbracket\varphi\rrbracket))$$
3.3 Axiomatising Standard Dynamic Models
Axiomatising Sequential Composition

Sequential composition axiom: \langle \alpha; \beta \rangle p \leftrightarrow \langle \alpha \rangle \langle \beta \rangle p.

Recall: \heartsuit : Q \Rightarrow Q \circ T \quad \overset{\text{1-1}}{\iff} \quad \hat{\heartsuit} : T \Rightarrow Q^{\text{op}} Q

Lemma (Soundness for sequential composition)
If \hat{\heartsuit} : T \Rightarrow Q^{\text{op}} Q is a monad morphism, and \gamma : X \rightarrow (TX)^A is \text{-standard}, then \langle \alpha; \beta \rangle p \leftrightarrow \langle \alpha \rangle \langle \beta \rangle p is valid in \gamma.

Note: Holds for \heartsuit iff holds for \neg \heartsuit \neg.
Examples

Remark: Monad morphism $T \Rightarrow Q^{\text{op}}Q$ \[\text{Eilenberg-Moore algebra } T^2 \rightarrow 2\]  

- **Kripke diamond** ($T = \mathcal{P}$):
  \[\Diamond : Q \Rightarrow Q\mathcal{P} \quad \text{corr. to} \quad \Diamond : \mathcal{P}\mathcal{P}(1) \rightarrow \mathcal{P}(1) \quad \text{(free } \mathcal{P}\text{-algebra)}\]
  so $\Diamond : \mathcal{P} \rightarrow Q^{\text{op}}Q$ is a monad morphism.

- **Monotonic nbhd diamond** ($T = \mathcal{M}$):
  \[\heartsuit : Q \Rightarrow Q\mathcal{M} \quad \text{corr. to} \quad \heartsuit : \mathcal{M} \Rightarrow Q^{\text{op}}Q \quad \text{(inclusion)}\]
  hence a monad morphism.
Axiomatising Pointwise Operations

- Example: PDL axiom for choice $[\alpha \cup \beta]p \leftrightarrow [\alpha]p \land [\beta]p$.
- Idea: $\hat{\Diamond}: T \Rightarrow N$ turns operation $\sigma$ on $T$ into operation $\chi$ on $N$.

\[
\begin{array}{c}
T^n \xrightarrow{\hat{\Diamond}^n} N^n \\
\downarrow \sigma \quad \downarrow \chi \\
T \xrightarrow{\hat{\Diamond}} N
\end{array}
\quad \text{For example:}

\[
\begin{array}{c}
\mathcal{P} \times \mathcal{P} \xrightarrow{\hat{\Box} \times \hat{\Box}} N \times N \\
\downarrow \cup \quad \downarrow \cap \\
\mathcal{P} \xrightarrow{\hat{\Box}} N
\end{array}
\]

- Need: $\chi: N^n \Rightarrow N$ such that the diagram commutes.
- From $\chi: N^n \Rightarrow N$, we get rank-1 formula $\varphi(\chi, \alpha_1, \ldots, \alpha_n, p)$ (details in paper).
- Def. $\varphi$ is rank 1 if $\varphi \in \text{Prop}(\Lambda(\text{Prop}(P_0)))$.
  Example: $\Box(p \rightarrow q) \rightarrow (\Box p \rightarrow \Box q)$
Lemma (Soundness)

If $\gamma: \mathcal{X} \to (TX)^A$ is $\theta$-standard, and there exists $\chi: \mathcal{N}^n \Rightarrow \mathcal{N}$ such that $\hat{\bigotimes} \circ \sigma = \chi \circ \hat{\bigotimes}^n$, then the rank-1 formula

$$\langle \sigma(\alpha_1, \ldots, \alpha_n) \rangle p \leftrightarrow \varphi(\chi, \alpha_1, \ldots, \alpha_n, p)$$

is valid in $\gamma$.

Positive operations

- $\chi: \mathcal{N}^n \Rightarrow \mathcal{N}$ corr. to $\check{\chi} \in \mathcal{N}(n \cdot Q(2))$, the free Boolean algebra on $n \cdot Q(2)$, i.e. $n$ copies of elements of $Q(2)$.
- Def. $\sigma$ is positive, if corresponding $\check{\chi}$ can be expressed without $\neg$. 
Axiomatising Tests

In PDL: \[ [\varphi?]\ p \leftrightarrow (\varphi \rightarrow p) \quad \text{or} \quad \langle \varphi? \rangle p \leftrightarrow (\varphi \land p) \]

In GL: \[ \langle \varphi? \rangle p \leftrightarrow (\varphi \land p) \]

Axioms (when \( \langle \alpha \rangle \) interpreted by \( \heartsuit \)):

- If \( \heartsuit \) is “box-like”,
  then add \( \langle \varphi? \rangle p \leftrightarrow (\varphi \rightarrow p) \) as frame condition.

- If \( \heartsuit \) is “diamond-like”,
  then add \( \langle \varphi? \rangle p \leftrightarrow (\varphi \land p) \) as frame condition.
Axiomatising Tests: $\Diamond$ is diamond/box-like

Let $\Diamond : Q \Rightarrow Q \circ T$ be a predicate lifting. We say that

- $\Diamond$ is diamond-like if for all sets $X$, all $U \subseteq X$, and all $\{t_i \mid i \in I\} \subseteq TX$:

$$\bigvee_{i \in I} t_i \in \Diamond_X(U) \iff \exists i \in I : t_i \in \Diamond_X(U).$$

- $\Diamond$ is box-like if for all sets $X$, all $U \subseteq X$, and all $\{t_i \mid i \in I\} \subseteq TX$:

$$\bigvee_{i \in I} t_i \in \Diamond_X(U) \iff \forall i \in I : t_i \in \Diamond_X(U).$$

Note: this has nothing to do with requiring the modality to preserve disjunctions or conjunctions!
Axiomatising Iteration (diamond-like \( \heartsuit \))

(Following GL axiomatisation)

For \( \alpha \in A \):

\[
\text{axiom: } \langle \alpha^* \rangle p \leftrightarrow p \lor \langle \alpha \rangle \langle \alpha^* \rangle p \quad \langle \alpha^* \rangle p \text{ is fixed point}
\]

\[
\text{rule: } p \lor \langle \alpha \rangle q \rightarrow q \quad \langle \alpha^* \rangle p \text{ is least prefixed point}
\]

\[
\frac{p \lor \langle \alpha \rangle q \rightarrow q}{\langle \alpha^* \rangle p \rightarrow q}
\]

Soundness over standard models: \( \checkmark \)
Logic and Derivability

Def. A modal logic $\mathcal{L} = (\Lambda, \text{Ax}, \text{Fr}, \text{Ru})$ consists of

- a modal signature $\Lambda$,
- a set of rank-1 axioms $\text{Ax} \subseteq \text{Prop}(\Lambda(\text{Prop}(P_0)))$,
- a set of frame conditions $\text{Fr} \subseteq \mathcal{F}(\Lambda, P_0)$,
- a set of inference rules $\text{Ru} \subseteq \mathcal{F}(\Lambda, P_0) \times \mathcal{F}(\Lambda, P_0)$.

Def. (Hilbert system derivability)

- $\vdash_{\mathcal{L}} \varphi$ if $\varphi$ is derivable from $\text{Ax} \cup \text{Fr}$ using propositional reasoning, uniform substitution, rules in $\text{Ru}$, and the congruence rule:

$$\frac{\varphi \leftrightarrow \psi}{\Box \varphi \leftrightarrow \Box \psi} \quad (\Box \in \Lambda)$$
Coalgebraic Dynamic Logic
(diamond-like $\Diamond$)

Given

- “base” logic $\mathcal{L} = (\{\Diamond\}, \text{Ax}(\Diamond, T), \emptyset, \emptyset)$ for $T$
- $\theta : \Sigma T \Rightarrow T$ and set $A_0$ of atomic actions.

we define the dynamic logic $\mathcal{L}(\theta, ;, *, ?) = (\Lambda, \text{Ax}, \text{Fr}, \text{Ru})$ by taking

\[
\begin{align*}
\Lambda &= \{\langle \alpha \rangle \mid \alpha \in A\}, \\
\text{Ax} &= \text{Ax}(\Diamond, T)_A \cup \text{"\theta-axioms"}, \\
\text{Fr} &= \{\langle \alpha; \beta \rangle p \leftrightarrow \langle \alpha \rangle \langle \beta \rangle p \mid \alpha, \beta \in A\} \\
&\quad \cup \{\langle \alpha^* \rangle p \leftrightarrow p \lor \langle \alpha \rangle \langle \alpha^* \rangle p \mid \alpha \in A\} \\
&\quad \cup \{\langle q? \rangle p \leftrightarrow (q \land p)\} \\
\text{Ru} &= \left\{ \frac{p \lor \langle \alpha \rangle q \rightarrow q}{\langle \alpha^* \rangle p \rightarrow q} \right\}
\end{align*}
\]
Strong Completeness for Iteration-Free Logics

Theorem

If base logic $\mathcal{L}$ satisfies conditions for quasi-canonical $T$-model [Schröder & Pattinson, 2009], then dynamic logic $\mathcal{L}(\theta, ;, ?)$ is sound and strongly complete wrt $\theta$-dynamic $T$-models.

- Strong completeness of PDL$\nabla \ast$ and GL$\nabla \ast$ recovered.
- Modest new result for “lift” monad $L(X) = 1 + X$.
Weak Completeness Needs Strong Coherence

- Standard weak completeness argument:
  1. build model on collection $S$ of finite maximally consistent sets of "relevant" formulas (needs the definition of closure - a generalised notion of a subformula)
  2. prove truth lemma for this finite model
- any subset $U \subseteq S$ of the constructed finite model can be characterised with a formula

$$\xi_U = \bigvee_{\Delta \in U} \bigwedge \Delta$$

Key for completeness: “strong coherence” property

We say that $\gamma: S \to (TS)^A$ is strongly coherent for $\alpha \in A$ if for all $\Gamma \in S$ and all $U \subseteq S$:

$$\hat{\gamma}(\alpha)(\Gamma) \in \bigvee S(U) \quad \text{iff} \quad \langle \alpha \rangle \xi_U \wedge \Gamma \text{ is } \mathcal{L}\text{-consistent.}$$
Weak Completeness with Iteration and Positive Operations

(Recall: PDL is not compact, hence not strongly complete.)

**Theorem**

If base logic $\mathcal{L}$ is one-step complete for $T$, and $\theta$ consists of positive operations, then $\mathcal{L}(\theta, ; , *, ?)$ is complete wrt standard, dynamic $T$-models.

- Completeness of PDL and dual-free GL recovered.
- Modest new result for dual-free GL with intersection.
Summary of Requirements

We require for semantics:

- $\mathbb{T} = (T, \eta, \mu)$, a Set-monad.
- $TX$ carries sup-lattice structure.
- $\heartsuit : Q \Rightarrow Q \circ T$ is predicate lifting for $T$.
- $\theta : \Sigma T \Rightarrow T$, a natural $\Sigma$-algebra.

We require for soundness and completeness:

- for each $\sigma : T^n \Rightarrow T$, a $\chi : N^n \Rightarrow N$ s.t. $\heartsuit \circ \sigma = \chi \circ \heartsuit^n$.
- $\heartsuit$ is monad morphism.
- $\heartsuit$ is monotone.
- Kleisli composition is monotone.
- $\theta$ consists of “positive operations” ($\checkmark$ negation-free)
Game Logic Automata

Joint work with:
Clemens Kupke, Johannes Marti, Yde Venema

Work in progress...
Game Logic, Completeness and $\mu$-Calculus

- Game Logic can be translated into (monotonic) $\mu$ML.
- Game Logic spans all levels of the alternation hierarchy $\mu$ML [Berwanger, 2003] (by interleaving $*$ and $d$)
- $\mu$ML completeness is hard...but automata can help.
Completeness via Modal Parity Automata

Completeness of the (coalgebraic) modal $\mu$-calculus
(Enqvist, Seifan, Venema)
Completeness via Modal Parity Automata

Completeness of the (coalgebraic) modal $\mu$-calculus (Enqvist, Seifan, Venema)
Modal Parity Automata for $\mu$ML

- Let $P$ be a set of atomic propositions, and
- $\text{Lit} = \{p, \neg p \mid p \in P\}$.
- For a set $S$, $1ML(P, S) = \text{Prop}(\text{Lit} \cup \{\Box t, \Diamond t \mid t \in \text{Latt}(S)\})$

**Def.** A modal parity automaton $A = (S, \Theta, \Omega, s_I)$ consists of

- a set $S$ of states,
- a 1-step transition structure $\Theta: S \to 1ML(P, S)$,
- a priority function $\Omega: S \to \omega$,
- an initial state $s_I \in S$.

Acceptance of Kripke structures defined via parity game.
Pre-Automata for Game Logic

Game Logic

- Both formulas and games must be “decomposed” (use reduction axioms).
- Assigning priorities to all vertices is problematic.

We work with pre-automata.

Def. A pre-automaton (or modal graph) $G = (V, E, L, \Omega)$ consists of

- a finite graph $(V, E)$,
- a labelling $L: V \to \Lit \cup \{\vee, \wedge\} \cup \{\langle \alpha \rangle, \langle \alpha^d \rangle | \alpha \in A_0\}$,
- a priority function $\Omega: S \to \omega$ where $S \subseteq V$

such that

- Arities match: if $L(v) \in \{\vee, \wedge\}$ then $|E(v)| \leq 2$, etc.
- On every cycle there is at least one state from $S$. 

Helle Hvid Hansen (TU Delft)
Pre-Automata for Game Logic

We have (to be checked and possibly tweaked):

- Construction from pre-automaton to automaton.
- Correspondence between GL formulas and game pre-automata.
- Conditions that characterise game pre-automata (conditions on cycles, sharing paths and maximal priorities)
- Evaluation game for pre-automata.
Conclusion

Summary:
- General coalgebraic completeness of PDL and dual-free Game Logic.
- Modest “new” results as instantiations.
- Need more examples...

Future Work:
- Extend to quantitative setting.
  Problem: for $T = D_\omega$ the distribution monad, the only EM-algebras $D_\omega(2) \to 2$ seem to be $\Diamond_{>0}$ and $\Diamond_{\leq 0}$.
  $\Rightarrow$ Switch to multi-valued logic.
- Extend to other types of operations (e.g. Coalition Logic).
- Completeness of full GL and coalgebraic CML.

THANKS!