Towards a Stochastic Interpretation of Game Logic

Ernst-Erich Doberkat
Overview

1 Modal Logics

2 Dynamic Logics
   - Aside: Banach-Mazur Games
   - Determined Games

3 Stochastic Interpretations
   - Modal Logics
   - Some Monads
   - Visit the Machine Room

4 Stochastic Nondeterminism
The formulas of a modal logic are given through the grammar

\[ \varphi ::= p \mid \varphi_1 \land \varphi_2 \mid \neg \varphi \mid \Diamond \varphi \]

Here \( p \in P \) is an atomic formula.

The usual conventions apply (e.g., \( \varphi_1 \lor \varphi_2 \) is written for \( \neg (\neg \varphi_1 \land \neg \varphi_2) \), \( \Box \varphi \) for \( \neg \Diamond \neg \varphi \), etc).

Intuitively

We say that \( \Diamond \varphi \) holds in state \( w \) iff we can make a transition from \( w \) to a state \( w' \) in which \( \varphi \) holds.

Kripke Model

A Kripke model \((W, V, R)\) for the interpretation of the logic is given by

- a set \( W \) of states (or worlds),
- a map \( V : P \to 2^W \), saying where the atomic formulas hold,
- a transition relation \( R \subseteq W \times W \).
VALIDITY

The validity relation $\models_K$ is defined recursively along the formula’s structure:

- $w \models_K p$ iff $w \in V(p)$, whenever $p \in P$,
- $w \models_K \varphi_1 \land \varphi_2$ iff $w \models_K \varphi_1$ and $w \models_K \varphi_2$,
- $w \models_K \neg \varphi$ iff $w \models_K \varphi$ is false,
- $w \models_K \Diamond \varphi$ iff there exists $w'$ with $\langle w, w' \rangle \in R$ such that $w' \models_K \varphi$.

This is what you would expect. One defines then morphisms of various kinds between models, has a look at expressivity (bisimilarity, logical and behavioral equivalence) through a relational or coalgebraic formulation, and in general have much fun with these models.
There is a more general way to interpret modal logics. For motivation, take a Kripke model \((W, V, R)\), put \(R(w) := \{w' \in W \mid \langle w, w' \rangle \in R\}\) as the set of successors to \(w \in W\). Look at

\[
M(w) := \{A \subseteq W \mid R(w) \cap A \neq \emptyset\}, \text{ (think of } \Diamond \text{)}
\]

\[
S(w) := \{A \subseteq W \mid R(w) \subseteq A\} \text{ (think of } \Box \text{)}.
\]

Thus, e.g., \(A \in M(w)\) iff \(A\) contains some states which can be reached from \(w\) via \(R\).

\(M(w)\) is an upper closed subset of \(\mathcal{P}(W)\): \(A \in M(w)\) and \(A \subseteq B\) together imply \(B \in M(w)\), similar for \(S(w)\).

A neighborhood model \((W, V, N)\) is defined just as a Kripke model, but \(N\) is a map from \(W\) to the upper closed subsets of \(\mathcal{P}(W)\).
$A \in N(w)$ means that $A$ contains the states which can be reached by $w$.

**Example**

Each Kripke model generates neighborhood models.

**Examples**

Let $Z(w)$ be the principal filter associated with $w$, i.e.,

$$A \in Z(w) \iff w \in A.$$

Then $(W, V, Z)$ is a neighborhood model.

Given a topological space $(W, \tau)$, let $U(w)$ be the neighborhood filter of $w$ with respect to $\tau$. Then $(W, V, U)$ constitutes a neighborhood model.
**Kripke models**

A Kripke model is based on a coalgebra for the power set functor $\mathcal{2}^-$. This functor is the functorial part of the power set monad, thus Kripke models are based on Kleisli morphisms for that monad.

**Neighborhood models**

A neighborhood model is based on a coalgebra for the “upper closed” functor $\mathcal{E} : W \mapsto \{ A \subseteq W \mid A \text{ is upper closed}\}$.

This functor is also the functorial part of a monad, thus neighborhood models are based on Kleisli morphisms for that monad.

**Recall**

A coalgebra $(A, f)$ for a functor $\mathcal{F}$ is an object $A$ together with a morphism $f : A \to \mathcal{F}(A)$. 
Definition

Given a neighborhood model $(W, V, N)$, define the validity sets $\llbracket \varphi \rrbracket$ for formulas $\varphi$ inductively:

- $\llbracket p \rrbracket := V(p)$ for $p \in P$,
- $\llbracket \varphi_1 \land \varphi_2 \rrbracket := \llbracket \varphi_1 \rrbracket \cap \llbracket \varphi_2 \rrbracket$,
- $\llbracket \neg \varphi \rrbracket := W \setminus \llbracket \varphi \rrbracket$,
- $\llbracket \Diamond \varphi \rrbracket := \{ w \in W \mid \llbracket \varphi \rrbracket \in N(w) \}$.

Put $w \models_N \varphi$ iff $w \in \llbracket \varphi \rrbracket$.

Hence

Thus $w \models_N \Diamond \varphi$ iff $\llbracket \varphi \rrbracket$ contains the states which can be reached from $w$. 

Visit the Machine Room
**Observation**

If the neighborhood model is generated from a Kripke model, then

\[ w \models_K \varphi \iff w \models_N \varphi. \]

Thus neighborhood models are more general than Kripke models.

They are even strictly more general, since there certainly exist neighborhood models which are not generated from Kripke models.
**INTRODUCE ACTIONS INTO THE LOGIC**

Let $A$ be a set of actions; we introduce a family $(\langle a \rangle)_{a \in A}$ of modal operators. The idea is that formula $\langle a \rangle \varphi$ holds in a world $w$ if action $a$ leads to a world $w'$ in which $\varphi$ holds. This is what the grammar now looks like:

$$\varphi ::= p \mid \varphi_1 \land \varphi_2 \mid \neg \varphi \mid \langle a \rangle \varphi$$

Here $p \in P$ is an atomic formula, and $a \in A$ is an action.

**MODIFY MODELS: Kripke**

Associate with each action $a$ a relation $R_a \subseteq W \times W$, and

$$w \models_K \langle a \rangle \varphi \iff R_a(w) \cap \llbracket \varphi \rrbracket \neq \emptyset.$$  

**MODIFY MODELS: Neighborhood**

Associate with each action $a$ an upper closed subset $N_a \subseteq 2^W$, and

$$w \models_N \langle a \rangle \varphi \iff \llbracket \varphi \rrbracket \in N_a(w).$$
The set $A$ of actions has been considered flat, i.e., without an inner structure. But sometimes the actions have a structure themselves. We look into

- Propositional Dynamic Logic (PDL)
- Game Logic

**The actions are programs**

A program $\pi$ is built up from primitive programs:

- compose $\pi_1; \pi_2$: execute first $\pi_1$, then $\pi_2$,
- choose $\pi_1 \cup \pi_2$: decide whether to branch into $\pi_1$ or $\pi_2$,
- iterate $\pi_1^*$: execute program $\pi_1$ a finite number of times, including not at all,
- test $\varphi?$: check whether or not property $\varphi$ holds.

E.g., $\varphi?; \pi_1 \cup (\neg \varphi)?; \pi_2$: If $\varphi$ is satisfied, execute $\pi_1$, otherwise, execute $\pi_2$. 
Thus a program $\pi$ is given by

$$\pi ::= t \mid \pi_1; \pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^* \mid \varphi?$$

with $t \in \Pi$ a primitive program, and $\varphi$ a formula of the underlying logic.

A Kripke model for PDL is given through $(W, V, (R_t)_{t \in \Pi})$ with $R_t$ a relation on $W$ for each primitive program. Thus we want to piece together the relations $R_\pi$ from the family $(R_t)_{t \in \Pi}$.

That’s not too bad, given that each $R_t$ is a Kleisli morphism for the power set monad, i.e., we have the Boolean operations, and a composition operator.
**Dynamic Logics**  
**PDL: Kripke**

**Here we go**

Define recursively

\[
R_{\pi_1 \cup \pi_2} := R_{\pi_1} \cup R_{\pi_2}, \\
R_{\pi_1 ; \pi_2} := R_{\pi_1} \circ R_{\pi_2}, \\
R_{\pi^*} := \bigcup_{n \geq 0} R_{\pi^n}
\]

**Test**

Postpone the definition of \( R_{\varphi} \) for the test operator. It depends on the semantics for the formulas.

**Clearly**

We have \( w \models_K \langle \pi \rangle \varphi \) iff there exists \( w' \in R_{\pi}(w) \) with \( w' \models_K \varphi \).
Transport this to neighborhood models. Let $N$ map $W$ to the upper closed subsets of $2^W$, define

$$N'(A) := \{ w \in W \mid A \in N(w) \}$$

for $A \subseteq W$.

Then $N' : 2^W \rightarrow 2^W$ is monotone. In fact: if $A \subseteq B$ and $A \in N(w)$, then $B \in N(w)$, since $N(w)$ is upper closed, hence $N'(A) \subseteq N'(B)$.

**Thus**

We can use the same machinery for neighborhood models, e.g.

$$N'_{\pi_1;\pi_2} := N'_{\pi_1} \circ N'_{\pi_2}.$$
Then

One can show $w \models_K \varphi \iff w \models_N \varphi$, if the neighborhood model is generated from a Kripke model. Thus the transition to neighborhood models is probably not worth the effort. Is it?

But

The picture changes once we have a look at Game Logics.

Introducing Angel and Demon

The game is played between Angel and Demon. They move in turn.
Assume we are in world $w$, and Angel plays game $\gamma$. We want to know what this may achieve. Hence we want to know which worlds Angel can reach by playing $\gamma$ in $w$.

This is certainly an upper closed set of subsets of $W$: If Angel can reach a world in $A$, and $A \subseteq B$, then Angel can reach a state in $B$ as well.

**Thus**

Hence we want to assign to each game $\gamma$ and each state $w \in W$ an upper closed subset $N'_\gamma(w) \subseteq 2^W$. 
Dynamic Logics
An aside

Questions
Are Angel and Demon going to win something? What is a strategy? Let us briefly look at Banach-Mazur games.

Banach-Mazur
A B-M game is played on \( \mathbb{N} \). Angel plays \( n_0 \). Depending on \( n_0 \), Demon plays \( n_1 \). Angel takes \( \langle n_0, n_1 \rangle \) into account and plays \( n_2 \), Demon reflects on \( \langle n_0, n_1, n_2 \rangle \) and counters with \( n_3 \), etc; the game never ends.
The actions are described through a trajectory in \( \mathbb{N}^\infty \).

Strategy
This permits defining strategies as maps \( \bigcup_{n \geq 0} \mathbb{N}^{2n} \to \mathbb{N} \) and \( \bigcup_{n \geq 0} \mathbb{N}^{2n+1} \to \mathbb{N} \) for Angel resp. Demon.
**Winning Strategies**

The goal of the B-M game $G_A$ is given by a subset $A \subseteq \mathbb{N}^\infty$. A strategy $\sigma$ for Angel is a **winning strategy** for Angel iff, no matter what Demon does, the trajectory of the game is in $A$, when Angel plays according to $\sigma$. Similarly for Demon (all trajectories must then be in $\mathbb{N}^\infty \setminus A$).

**Determined Games**

B-M game $G_A$ is called **determined** iff either Angel or Demon has a winning strategy.

**Axiom of Determinacy**

Each B-M game $G_A$ is determined.

Looks a bit far fetched . . .
**Dynamic Logics**

**An aside**

---

**We know**

The Axiom of Determinacy is incompatible with the Axiom of Choice.

**Axiom of Choice**

There exists a non-measurable subset of $[0, 1]$.

**Axiom of Determinacy**

Every subset of $[0, 1]$ is measurable.

---

**What would Hamlet do?**

The Axiom of Choice seems to be somewhat indispensable, but, on the other hand, the Axiom of Determinacy is also somewhat practical.
**Returning to Game Logics**

The modalities in Game Logics are games.

**Games**

Games are built up from primitive games in this way:

\[ \gamma ::= g \mid \gamma^d \mid \gamma_1; \gamma_2 \mid \gamma_1 \cup \gamma_2 \mid \gamma_1 \cap \gamma_2 \mid \gamma^* \mid \gamma^\times \mid \psi \]

Here \( g \in \Gamma \) is a *primitive game*, and \( \psi \) a formula of the underlying logic.

Composition, \( \gamma_1 \cup \gamma_2 \) and \( \gamma^* \) are as in PDL, \( \gamma_1 \cap \gamma_2 \) is *demonic* choice, \( \gamma^\times \) is *demonic* iteration. \( \gamma^d \) is *demonization*: Angel and Demon changes places.

**PDL**

Propositional Dynamic Logics is a fragment of Game Logics (just don’t use the operators associated with Demon).
Angel and Demon play against each other. If the objective for Angel is to achieve a state in which a formula $\varphi$ holds, the objective for Demon is to achieve a state in which $\neg \varphi$ is true.

**Strategy?**

It is assumed that Angel and Demon follow some strategy. We do not say formally, however, what a strategy is (in contrast to B-M games), but rest on an informal understanding.

**Determinedness**

We assume that the game is determined in this sense: If Angel does not have a strategy for achieving a formula $\varphi$, Demon has a strategy for achieving $\neg \varphi$, and vice versa.
**Dynamic Logics**

**Determineness: Consequences**

---

### Some Consequences

It is basically sufficient to model Angel’s behavior: \( \gamma_1 \cap \gamma_2 \) is equivalent to \((\gamma_1^d \cup \gamma_2^d)^d\), and \( \gamma^\times \) is equivalent to \((\gamma^d)^*^d\).

---

### Neighborhood Model

The interpretation of game logics will be done through a neighborhood model \((W, V, (N_g)_{g \in \Gamma})\). Thus we have for each primitive game \( g \in G \) and any world \( w \in W \) an upper closed subset \( N_g(w) \subseteq 2^W \).

---

### Transform

Put \( N'_g(A) := \{ w \in W \mid A \in N_g(w) \} \). Then \( N'_g(A) \) is the set of states which Angel can achieve when it plays primitive game \( g \in \Gamma \) in a state taken from \( A \).
It is convenient to assign to each game $\gamma$ a map $N'_\gamma : \mathcal{P}(W) \to \mathcal{P}(W)$, as we did in the model case PDL. We basically know from PDL how Angel operates by transporting the algebraic operations of programs to maps $\mathcal{P}(W) \to \mathcal{P}(W)$.

**Demonization**

Define for $N : \mathcal{P}(W) \to \mathcal{P}(W)$ its demonization

$$\partial N : A \mapsto W \setminus N(W \setminus A).$$

If $N'_\gamma : \mathcal{P}(W) \to \mathcal{P}(W)$ is defined for game $\gamma$, define $N'_{\gamma,d} := \partial N'_\gamma$.

**Interpretation**

We define in this manner inductively a map $N_\gamma : \mathcal{P}(W) \to \mathcal{P}(W)$ for each game $\gamma$. Then we set

$$[[\langle \gamma \rangle \varphi]] := N'_\gamma([[\varphi]]).$$
**Dynamic Logics**

Kripke Models?

---

**But ...**

This is formulated for neighborhood models. Why don’t we take Kripke models for interpreting game logics?

---

Each Kripke model spawns a neighborhood model, so Kripke models are not excluded.

---

**There is a catch, though**

It can be shown that games are distributive under the class of Kripke models, i.e.,

\[ w \models_{K} \langle \gamma; (\gamma_1 \cup \gamma_2) \rangle \varphi \iff w \models_{K} (\langle \gamma; \gamma_1 \rangle \cup \langle \gamma; \gamma_2 \rangle) \varphi \]

holds.

---

This is clearly inadequate.
Changes ahead

We want to say in a probabilistic setting that a formula holds with a probability of at most 70% after performing some action. Hence our modal logics needs to be modified.

Modal logics

Formulas now look like this

\[ \varphi ::= p \mid \varphi_1 \land \varphi_2 \mid \neg \varphi \mid \langle a : r \rangle \varphi \]

with — again — \( p \in P \) an atomic formula, \( a \in A \) an action, and \( r \in [0, 1] \) as a kind of threshold value.

The informal understanding is that formula \( \langle a : r \rangle \varphi \) holds in a world \( w \) iff the probability for transition from \( w \) after action \( a \) to a state in which \( \varphi \) holds is at least \( r \).
Stochastic Interpretation
Entering the realm of probabilities

**Interpretation?**

The interpretation of this logic is done for general probabilities. Thus we do not stick to probability distributions over finite or countable sets.

**Recap**

We assume that the set of worlds $W$ is a measurable space. This is a set with a Boolean $\sigma$-algebra on it, the members are called events. A probability $\mu$ assigns to each event $A$ its probability $\mu(A) \in [0, 1]$ with

- $\mu(\text{impossible event}) = \mu(\emptyset) = 0$, $\mu(\text{certain event}) = \mu(W) = 1$,
- $\mu(A \cup B) = \mu(A) + \mu(B)$, provided the events $A$ and $B$ are disjoint,
- if $A_1 \subseteq A_2 \subseteq \ldots$, then $\mu(A_1) \leq \mu(A_2) \leq \ldots$ and $\mu(\bigcup_{n \geq 1} A_n) = \sup_{n \in \mathbb{N}} \mu(A_n)$. 

Recap
Actually
The set $(W)$ of all probabilities on $W$ is a measurable space itself.

But we know more
$\mathcal{S}$ is an endofunctor on the category of all measurable spaces. It is the functorial part of a monad (which is sometimes called the Giry monad).

But we know still more
A stochastic relation $K$ on $W$ assigns to each $w \in W$ a probability $K(w) \in \mathcal{S}(W)$. $K(w)(A)$ is interpreted as the probability for a transition starting from $w$ hitting an element of event $A$. These are exactly the Kleisli morphisms for the Giry monad.

Stochastic relations are also known as Markov kernels or transition probabilities.
**Stochastic Interpretation**

**Stochastic Kripke Models**

**Model**

We interpret this kind of modal logic through a stochastic Kripke model $(W, V, (K_a)_{a \in A})$. Now $V$ maps the atomic formulas to the measurable sets (thus $V(p)$ becomes an event), and $K_a$ is a stochastic relation on $W$ for each $a \in A$.

$K_a(w)(A)$ is the probability that an $a$-transition from $w$ ends up in an element of $A$.

**Interpretation**

The interpretation of the Boolean cases remains as it is, and

$$w \models K \langle a :: r \rangle \varphi \iff K_a(w)([\varphi]) \geq r.$$ 

Thus $\langle a :: r \rangle \varphi$ holds in $w$ iff we can make an $a$-transition from $w$ into a state in which $\varphi$ holds with probability at least $r \in [0, 1]$.

**Remark**

Actually, we can do without negation in the logic, since

$$w \models K \neg \varphi \iff w \not\in [\varphi].$$
<table>
<thead>
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2^E</td>
<td>E^E</td>
<td>$^E</td>
</tr>
<tr>
<td>Logic General modal logic.</td>
<td>Logic General modal logic, game logic</td>
<td>Logic Stochastic modal logic.</td>
</tr>
</tbody>
</table>

**Uniform picture**

The model is based in each case on a family of Kleisli morphisms for the corresponding monad. This leads to *coalgebraic logic*. 
**STOCHASTIC INTERPRETATION**

**Dynamic Logics**

---

**Requirements**

A stochastic interpretation of a dynamic logic will have to deal with probabilities over the set of worlds (rather than the set of worlds proper).

\[
\text{state} \implies \text{distribution over states} \\
\text{achievable} \implies \text{achievable events} \\
\text{states} \implies \text{of state probabilities}
\]

---

**Basic construction**

\(P_\gamma(w)\) is an upper closed subset of events over probability distributions on \(W\) for each \(w \in W\). Then \(G \in P_\gamma(w)\) means: Angel has a strategy for achieving a probability in \(G\) as a distribution for the next state, if it plays game \(\gamma\) in state \(w\).

---

**In particular**

If \(\{\mu \in \$ (W) \mid \mu(A) \geq r\} \in P_\gamma(w)\), then the next state will be a member of event \(A\) with probability not less than \(r\).
**Q:** Can’t we compose, then, the upper closed monad with the Giry monad?

**A:** No, we can’t.

**Why?**
The composition of two monads is not necessarily a monad again (You need a lot of machinery, i.e., natural transformations, to connect them). That’s too bad.

**Now, what?**
We basically have to simulate the properties of a monad through a suitable functor. The composition of coalgebra morphisms is particularly important.

**Parental guidance suggested**
This may be suitable for mature audiences only, since there may be complications of a technical nature. I’ll give you the Disney version, all cleaned up, and a bit distorted.
Stochastic Interpretation
Examples

Take a model \((W, V, (P_g)_{g \in \Gamma})\), where \(P_g(w)\) is an upper closed subset of events over \(W\) for simple games \(g\).

Angellic Choice for Simple Games

\[ w \models ((g_1 \cup g_2) :: r) \varphi \iff \begin{cases} \{ \mu \mid \mu([\varphi]) \geq a_1 \} \in P_{g_1}(w), \\
\text{or} \\
\{ \mu \mid \mu([\varphi]) \geq a_2 \} \in P_{g_2}(w), \\
\text{for all rational } a_1, a_2 \text{ with } a_1 + a_2 < r. \end{cases} \]

for \(g_1, g_2 \in \Gamma\).

Demonization

Let \(g \in \Gamma\), then \(w \models (g^d :: r) \varphi\) iff \(\{ \mu \mid \mu([\varphi]) \geq r \} \in \partial P_g([\varphi])\).
Composition of two games is crucial (actually, it is exactly here that the composition of Kleisli morphisms would be very helpful). We want to define \( w \models \langle (g_1; g_2) :: r \rangle \varphi \) for \( g_1, g_2 \in \Gamma \).

**Intermediary step**

Let

\[
\mu \in H_{g_2}(\llbracket \varphi \rrbracket, q) \iff \int_0^1 \mu(\{w \mid w \models \langle g_2 :: r \rangle \varphi \}) \, dr \geq q.
\]

Then \( \mu \in H_{g_2}(\llbracket \varphi \rrbracket, q) \) iff we can expect for distribution \( \mu \) a \( g_2 \)-transition to lead into \( \llbracket \varphi \rrbracket \) with probability at least \( q \). It is these guys we should be looking for.

**Composition**

\[
w \models \langle (g_1; g_2) :: r \rangle \varphi \iff H_{g_2}(\llbracket \varphi \rrbracket, r) \in P_{g_1}(w).
\]
**Stochastic Interpretation**

To cut a long story short

**Induction over the syntactical structure**

In this way, we construct step by step the set

\[ [[\langle \gamma :: r \rangle \varphi]] \]

as the set of all states in which this formula holds, for all games \( \gamma \) and all formulas \( \varphi \).

**Theorem**

\[ [[\langle \gamma :: r \rangle \varphi]] \text{ is an event, provided the measurable space is complete.} \]

The semantics of iteration, i.e. for \( \langle \gamma^* :: r \rangle \varphi \), requires Boolean operations over an uncountable set, Boolean \( \sigma \)-algebras are usually only closed under countable operations. These \( * \)-operations are, however, well-behaved, and complete measurable spaces are closed under them (Souslin closure).
**Special case: Kripke model**

Each stochastic relation $K$ on $W$ gives an upper closed subset $P_K(w)$ of events over $W$ for each world $w$:

$$P_K(w) := \{ A \subseteq \mathcal{S}(W) \mid K(w) \in A \}.$$ 

Thus each Kripke model yields a stochastic model for Game Logic.

**Theorem (cp. D. Kozen)**

In a Kripke generated model, the semantics for the PDL fragment is the same as the semantics for PDL through the Kripke model.

**Remark**

Game Logic is distributive in a Kripke generated model.
This is a model for stochastic nondeterminism:

- nondeterministic choice of different possibilities,
- the objects to choose from are probability distributions.

We need to impose some additional structure for modelling the desired structures, in particular composition.

**On the monadic level**

Nondeterminism + randomness $\not\Rightarrow$ stochastic nondeterminism. This is so since we cannot compose the corresponding monads to obtain another monad.

**P. d’Argenio and P. Sánchez Terraf (MCS)**

Hit measurability of maps into the power set of $\mathcal{W}$. Interesting results for expressivity.

**P. Sánchez Terraf and EED. (JLC)**

Find bridges between these approaches.