Neighborhood Semantics for Modal Logic

Lecture 5

Eric Pacuit

ILLC, Universiteit van Amsterdam
staff.science.uva.nl/~epacuit

August 17, 2007

- ✓ Introduction, Motivation and Background Information
- ✓ Basic Concepts, Non-normal Modal Logics, Completeness, Incompleteness, Relation with Relational Semantics
- ✓ Decidability/Complexity, Related Semantics: Topological Semantics for Modal Logic, More on the Relation with Relational Semantics, Subset Models, First-order Modal Logic
- ✓ Some Model Theory: Monotonic Modal Logic, Model Constructions, First-Order Correspondent Language

Lecture 5: Neighborhood Semantics in Action: Game Logic, Coalgebra, Common Knowledge

What is the relationship between Neighborhood and other Semantics for Modal Logic? What about First-Order Modal Logic?

Can we import results/ideas from model theory for modal logic with respect to Kripke Semantics/Topological Semantics?

Monotonic Bisimulation

Let $\mathfrak{M} = \langle W, N, V \rangle$ and $\mathfrak{M}' = \langle W', N', V' \rangle$ be two monotonic neighborhood models. A relation $Z \subseteq W \times W'$ is a bisimulation provided whenever wZw':

Atomic harmony: for each $p \in \mathsf{At}$, $w \in V(p)$ iff $w' \in V'(p)$

Zig: If $X \in N(w)$ then there is an $X' \subseteq W'$ such that

$$X' \in N'(w') \text{ and } \forall x' \in X' \ \exists x \in X \text{ such that } xZx'$$

Zag: If $X' \in N'(w')$ then there is an $X \subseteq W$ such that

$$X \in N(w) \text{ and } \forall x \in X \ \exists x' \in X' \text{ such that } xZx'$$

Monotonic Bisimulation

Let $\mathfrak{M} = \langle W, N, V \rangle$ and $\mathfrak{M}' = \langle W', N', V' \rangle$ be two monotonic neighborhood models. A relation $Z \subseteq W \times W'$ is a bisimulation provided whenever wZw':

Atomic harmony: for each $p \in \mathsf{At}$, $w \in V(p)$ iff $w' \in V'(p)$

Zig: If $X \in N(w)$ then there is an $X' \subseteq W'$ such that

$$X' \in N'(w') \text{ and } \forall x' \in X' \; \exists x \in X \text{ such that } xZx'$$

Zag: If $X' \in N'(w')$ then there is an $X \subseteq W$ such that

$$X \in N(w) \text{ and } \forall x \in X \; \exists x' \in X' \text{ such that } xZx'$$

Lemma

On locally core-finite models, if $\mathbb{M}, w \leftrightsquigarrow \mathbb{M}', w'$ *then* $\mathbb{M}, w \,\underline{\leftrightarrow}\, \mathbb{M}', w'$.

Monotonic Bisimulation

Let $\mathfrak{M} = \langle W, N, V \rangle$ and $\mathfrak{M}' = \langle W', N', V' \rangle$ be two monotonic neighborhood models. A relation $Z \subseteq W \times W'$ is a bisimulation provided whenever wZw':

Atomic harmony: for each $p \in \mathsf{At}$, $w \in V(p)$ iff $w' \in V'(p)$

Zig: If $X \in N(w)$ then there is an $X' \subseteq W'$ such that

$$X' \in N'(w') \text{ and } \forall x' \in X' \; \exists x \in X \text{ such that } xZx'$$

Zag: If $X' \in N'(w')$ then there is an $X \subseteq W$ such that

$$X \in N(w) \text{ and } \forall x \in X \; \exists x' \in X' \text{ such that } xZx'$$

Theorem

A $\mathcal{L}_2$ formula $\alpha(x)$ is invariant for monotonic bisimulation, then $\alpha(x)$ is equivalent to $st_x^{mon}(\varphi)$ for some $\varphi \in \mathcal{L}$.

Monotonic Bisimulation

Let $\mathfrak{M} = \langle W, N, V \rangle$ and $\mathfrak{M}' = \langle W', N', V' \rangle$ be two monotonic neighborhood models. A relation $Z \subseteq W \times W'$ is a bisimulation provided whenever wZw':

Atomic harmony: for each $p \in \mathsf{At}$, $w \in V(p)$ iff $w' \in V'(p)$

Zig: If $X \in N(w)$ then there is an $X' \subseteq W'$ such that

$$X' \in N'(w') \text{ and } \forall x' \in X'\ \exists x \in X \text{ such that } xZx'$$

Zag: If $X' \in N'(w')$ then there is an $X \subseteq W$ such that

$$X \in N(w) \text{ and } \forall x \in X\ \exists x' \in X' \text{ such that } xZx'$$

M. Pauly. *Bisimulation for Non-normal Modal Logic*. 1999.

H. Hansen. *Monotonic Modal Logic*. 2003.

Do monotonic bisimulations work when we drop monotonicity? No!

Definition

Two points w_1 from $\mathfrak{F}_1$ and w_2 from $\mathfrak{F}_2$ are behavorially equivalent provided there is a neighborhood frame $\mathfrak{F}$ and bounded morphisms $f : \mathfrak{F}_1 \to \mathfrak{F}$ and $g : \mathfrak{F}_2 \to \mathfrak{F}$ such that $f(w_1) = g(w_2)$.

Theorem

Over the class **N** *(of neighborhood models), the following are equivalent:*

- $\alpha(x)$ *is equivalent to the translation of a modal formula*
- $\alpha(x)$ *is invariant under behavioural equivalence.*

H. Hansen, C. Kupke and EP. *Bisimulation for Neighborhood Structures*. CALCO 2007.

The Language $\mathcal{L}_2$

The language $\mathcal{L}_2$ is built from the following grammar:

$$x = y \mid u = v \mid \mathsf{P}_i x \mid x\mathsf{N}u \mid u\mathsf{E}x \mid \neg\varphi \mid \varphi \wedge \psi \mid \exists x\varphi \mid \exists u\varphi$$

$\mathfrak{M} = \langle D, \{P_i \mid i \in \omega\}, N, E\rangle$ where

- $D = D^{\mathsf{s}} \cup D^{\mathsf{n}}$ (and $D^{\mathsf{s}} \cap D^{\mathsf{n}} = \emptyset$),
- $Q_i \subseteq D^{\mathsf{s}}$,
- $N \subseteq D^{\mathsf{s}} \times D^{\mathsf{n}}$ and
- $E \subseteq D^{\mathsf{n}} \times D^{\mathsf{s}}$.

The Language $\mathcal{L}_2$

Definition

Let $\mathfrak{M} = \langle S, N, V\rangle$ be a neighbourhood model. The *first-order translation* of $\mathcal{M}$ is the structure $\mathfrak{M}^\circ = \langle D, \{P_i \mid i \in \omega\}, R_N, R_\ni\rangle$ where

- $D^{\mathsf{s}} = S$, $D^{\mathsf{n}} = \bigcup_{s\in S} N(s)$
- For each $i \in \omega$, $P_i = V(p_i)$
- $R_N = \{(s, U) \mid s \in D^{\mathsf{s}}, U \in N(s)\}$
- $R_\ni = \{(U, s) \mid s \in D^{\mathsf{s}}, s \in U\}$

The Language $\mathcal{L}_2$

Definition

The *standard translation* of the basic modal language are functions $st_x : \mathcal{L} \to \mathcal{L}_2$ defined as follows as follows: $st_x(p_i) = \mathsf{P}_i x$, st_x commutes with boolean connectives and

$$st_x(\Box\varphi) = \exists u(x\mathsf{R}_N u \wedge (\forall y(u\mathsf{R}_\ni y \leftrightarrow st_y(\varphi)))$$

The Language $\mathcal{L}_2$

Definition

The *standard translation* of the basic modal language are functions $st_x : \mathcal{L} \rightarrow \mathcal{L}_2$ defined as follows as follows: $st_x(p_i) = \mathsf{P}_i x$, st_x commutes with boolean connectives and

$$st_x(\Box\varphi) = \exists u(x\mathsf{R}_N u \wedge (\forall y(u\mathsf{R}_\ni y \leftrightarrow st_y(\varphi)))$$

Lemma

Let $\mathfrak{M}$ be a neighbourhood structure and $\varphi \in \mathcal{L}$. For each $s \in S$, $\mathfrak{M}, s \models \varphi$ iff $\mathfrak{M}^\circ \models st_x(\varphi)[s]$.

$\mathbf{N} = \{\mathfrak{M} \mid \mathfrak{M} \cong \mathfrak{M}^\circ$ for some neighbourhood model $\mathfrak{M}\}$

(A1) $\exists x(x = x)$

(A2) $\forall u \exists x(x \mathsf{R}_N u)$

(A3) $\forall u, v(\neg(u = v) \rightarrow$
$\exists x((u\mathsf{R}_\ni x \wedge \neg v\mathsf{R}_\ni x) \vee (\neg u\mathsf{R}_\ni x \wedge v\mathsf{R}_\ni x)))$

Theorem

Suppose $\mathfrak{M}$ is an $\mathcal{L}_2$-structure. Then there is a neighbourhood structure $\mathfrak{M}_\circ$ such that $\mathfrak{M} \cong (\mathfrak{M}_\circ)^\circ$.

Theorem

Over the class **N** *(of neighborhood models), the following are equivalent:*

- $\alpha(x)$ *is equivalent to the translation of a modal formula*
- $\alpha(x)$ *is invariant under behavioural equivalence.*

H. Hansen, C. Kupke and EP. *Bisimulation for Neighborhood Structures*. CALCO 2007.

What can we infer from the fact that bi-modal normal modal logic can simulate non-normal modal logics?

What can we infer from the fact that bi-modal normal modal logic can simulate non-normal modal logics?

Can we read off a notion of bisimulation?

What can we infer from the fact that bi-modal normal modal logic can simulate non-normal modal logics?

Can we read off a notion of bisimulation? Not clear.

What can we infer from the fact that bi-modal normal modal logic can simulate non-normal modal logics?

Can we read off a notion of bisimulation? Not clear.

- Decidability of the satisfiability problem
- Canonicity
- Salqhvist Theorem
- ????

O. Gasquet and A. Herzig. *From Classical to Normal Modal Logic*. .

M. Kracht and F. Wolter. *Normal Monomodal Logics can Simulate all Others* . .

H. Hansen (Chapter 10). *Monotonic Modal Logics*. 2003.

Theorem The McKinsey Axiom is canonical with respect to neighborhood semantics.

T. Surendonk. *Canonicty for Intensional Logics with Even Axioms*. JSL 2001.

Tableaux

There are tableaux for non-normal modal logics.

H. Hansen. *Tableau Games for Coalition Logic and Alternating-Time Temporal Logic*. 2004.

G. Governatori and A. Luppi. *Labelled Tableaux for Non-Normal Modal Logics*. Advances in AI (2000).

Tableaux

There are tableaux for non-normal modal logics.

The tableau rule for **K**:

$$\frac{\Phi \circ \Box\psi, \Psi}{\Phi^{\#} \circ \psi}$$

where $\Phi^{\#} = \{\varphi \mid \Box\varphi \in \Phi\}$

H. Hansen. *Tableau Games for Coalition Logic and Alternating-Time Temporal Logic*. 2004.

G. Governatori and A. Luppi. *Labelled Tableaux for Non-Normal Modal Logics*. Advances in AI (2000).

Tableaux

There are tableaux for non-normal modal logics.

The tableau rule for **EM**:

$$\frac{\Phi, \Box\varphi \circ \Box\psi, \Psi}{\varphi \circ \psi}$$

H. Hansen. *Tableau Games for Coalition Logic and Alternating-Time Temporal Logic*. 2004.

G. Governatori and A. Luppi. *Labelled Tableaux for Non-Normal Modal Logics*. Advances in AI (2000).

Algebra

- Categories of relational frames are dual to categories of Boolean algebras with *normal* operators. (arrows are bounded morphisms)

Y. Venema. *Algebras and Coalgebras*. Handbook of Modal Logic (2006).

Algebra

- Categories of relational frames are dual to categories of Boolean algebras with *normal* operators. (arrows are bounded morphisms)
- Categories of neighborhood frames are dual to categories of Boolean algebras with arbitrary operators.

Kosta Dosen. *Duality Between Modal Algebras and Neighborhood Frames*. Studia Logica (1987).

Y. Venema. *Algebras and Coalgebras*. Handbook of Modal Logic (2006).

Coalgebra

A Coalgebra for a functor F in a category C is a pair $\langle A, \alpha \rangle$ where A is an object of C and $\alpha : A \to \mathsf{F}A$.

Y. Venema. *Algebras and Coalgebras*. Handbook of Modal Logic (2006).

Coalgebra

A Coalgebra for a functor F in a category C is a pair $\langle A, \alpha \rangle$ where A is an object of C and $\alpha : A \to \mathsf{F}A$.

Examples:

- $C \times X$: streams over C
- $2 \times X^C$: deterministic automata with input alphabet C
- $\wp(C \times X)$: labeled transition systems
- $(1 + \Delta(X))^C$: probabilistic transition systems

Y. Venema. *Algebras and Coalgebras*. Handbook of Modal Logic (2006).

Coalgebra

A Coalgebra for a functor F in a category C is a pair $\langle A, \alpha \rangle$ where A is an object of C and $\alpha : A \to \mathsf{F}A$.

The contravariant power set functor 2 is the functor that maps a set X to the set $\wp(X)$ of subsets of X and a function $f : X \longrightarrow Y$ to the inverse image functions $f^{-1} : \wp(Y) \to \wp(X)$ given by $f^{-1}[U] := \{x \in X \mid f(x) \in U\}$.

The functor 2^2 is defined as the composition $2 \circ 2$ of 2 with itself.

Neighbourhood frames are coalgebras for the functor 2^2.

Y. Venema. *Algebras and Coalgebras.* Handbook of Modal Logic (2006).

Common Belief/Knowledge in Non-Normal Modal Logics

It is possible to separate the two definitions of common belief (knowledge) as a least fixed point and infinite iteration using neighborhood models.

Common Belief/Knowledge in Non-Normal Modal Logics

It is possible to separate the two definitions of common belief (knowledge) as a least fixed point and infinite iteration using neighborhood models.

There are non-normal logics with a common belief (knowledge) operator that are sound and **strongly** complete.

A. Heifetz. *Common belief in monotonic epistemic logic*. Mathematical Social Sciences (1999).

Lismont and Mongin. *Strong Completeness Theorems for Weak Logics of Common Beliefs*. JPL (2003).

J. van Benthem and D. Saraenac. *Geometry of Knowledge*. 2004.

Background: Propositional Dynamic Logic

Let P be a set of atomic programs and At a set of atomic propositions.

Formulas of **PDL** have the following syntactic form:

$$\varphi := p \mid \bot \mid \neg\varphi \mid \varphi \vee \psi \mid [\alpha]\varphi$$

$$\alpha := a \mid \alpha \cup \beta \mid \alpha;\beta \mid \alpha^* \mid \varphi?$$

where $p \in$ At and $a \in$ P.

$[\alpha]\varphi$ is intended to mean "after executing the program α, φ is true"

Background: Propositional Dynamic Logic

Semantics: $\mathcal{M} = \langle W, \{R_a \mid a \in \mathsf{P}\}, V\rangle$ where for each $a \in \mathsf{P}$, $R_a \subseteq W \times W$ and $V : \mathsf{At} \to \wp(W)$

- $R_{\alpha\cup\beta} := R_\alpha \cup R_\beta$
- $R_{\alpha;\beta} := R_\alpha \circ R_\beta$
- $R_{\alpha^*} := \cup_{n\geq 0} R_\alpha^n$
- $R_{\varphi?} = \{(w, w) \mid \mathcal{M}, w \models \varphi\}$

$$\mathcal{M}, w \models [\alpha]\varphi \text{ iff for each } v, \text{ if } wR_\alpha v \text{ then } \mathcal{M}, v \models \varphi$$

Background: Propositional Dynamic Logic

1. Axioms of propositional logic
2. $[\alpha](\varphi \rightarrow \psi) \rightarrow ([\alpha]\varphi \rightarrow [\alpha]\psi)$
3. $[\alpha \cup \beta]\varphi \leftrightarrow [\alpha]\varphi \wedge [\beta]\varphi$
4. $[\alpha; \beta]\varphi \leftrightarrow [\alpha][\beta]\varphi$
5. $[\psi?]\varphi \leftrightarrow (\psi \rightarrow \varphi)$
6. $\varphi \wedge [\alpha][\alpha^*]\varphi \leftrightarrow [\alpha^*]\varphi$
7. $\varphi \wedge [\alpha^*](\varphi \rightarrow [\alpha]\varphi) \rightarrow [\alpha^*]\varphi$
8. Modus Ponens and Necessitation (for each program α)

Background: Propositional Dynamic Logic

1. Axioms of propositional logic
2. $[\alpha](\varphi \rightarrow \psi) \rightarrow ([\alpha]\varphi \rightarrow [\alpha]\psi)$
3. $[\alpha \cup \beta]\varphi \leftrightarrow [\alpha]\varphi \wedge [\beta]\varphi$
4. $[\alpha; \beta]\varphi \leftrightarrow [\alpha][\beta]\varphi$
5. $[\psi?]\varphi \leftrightarrow (\psi \rightarrow \varphi)$
6. $\varphi \wedge [\alpha][\alpha^*]\varphi \leftrightarrow [\alpha^*]\varphi$ (Fixed-Point Axiom)
7. $\varphi \wedge [\alpha^*](\varphi \rightarrow [\alpha]\varphi) \rightarrow [\alpha^*]\varphi$ (Induction Axiom)
8. Modus Ponens and Necessitation (for each program α)

Background: Propositional Dynamic Logic

Theorem PDL is sound and weakly complete with respect to the Segerberg Axioms.

Theorem The satisfiability problem for **PDL** is decidable (EXPTIME-Complete).

D. Kozen and R. Parikh. *A Completeness proof for Propositional Dynamic Logic*. .

D. Harel, D. Kozen and Tiuryn. *Dynamic Logic*. 2001.

Concurrent Programs

D. Peleg. *Concurrent Dynamic Logic.* JACM (1987).

Concurrent Programs

$\alpha \cap \beta$ is intended to mean "execute α and β in parallel".

D. Peleg. *Concurrent Dynamic Logic*. JACM (1987).

Concurrent Programs

$\alpha \cap \beta$ is intended to mean "execute α and β in parallel".

In PDL: $R_\alpha \subseteq W \times W$, where $wR_\alpha v$ means executing α in state w leads to state v.

D. Peleg. *Concurrent Dynamic Logic.* JACM (1987).

Concurrent Programs

$\alpha \cap \beta$ is intended to mean "execute α and β in parallel".

In PDL: $R_\alpha \subseteq W \times W$, where $wR_\alpha v$ means executing α in state w leads to state v.

With Concurrent Programs: $R_\alpha \subseteq W \times \wp(W)$, where $wR_\alpha V$ means executing α in parallel from state w to reach all states in V.

D. Peleg. *Concurrent Dynamic Logic.* JACM (1987).

Concurrent Programs

$\alpha \cap \beta$ is intended to mean "execute α and β in parallel".

In PDL: $R_\alpha \subseteq W \times W$, where $wR_\alpha v$ means executing α in state w leads to state v.

With Concurrent Programs: $R_\alpha \subseteq W \times \wp(W)$, where $wR_\alpha V$ means executing α in parallel from state w to reach all states in V.

$w \models \langle\alpha\rangle\varphi$ iff $\exists U$ such that $(w, U) \in R_\alpha$ and $\forall v \in U$, $v \models \varphi$.

$R_{\alpha\cap\beta} := \{(w, V) \mid \exists U, U',\ (w, U) \in R_\alpha,\ (w, U') \in R_\beta,\ V = U \cup U'\}$

D. Peleg. *Concurrent Dynamic Logic*. JACM (1987).

From **PDL** to Game Logic

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

From **PDL** to Game Logic

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

Main Idea:
In **PDL**: $w \models \langle \pi \rangle \varphi$: there is a run of the program π starting in state w that ends in a state where φ is true.

The programs in **PDL** can be thought of as *single player games*.

From **PDL** to Game Logic

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

Main Idea:

In **PDL**: $w \models \langle\pi\rangle\varphi$: there is a run of the program π starting in state w that ends in a state where φ is true.

The programs in **PDL** can be thought of as *single player games*.

Game Logic generalized **PDL** by considering two players:

In **GL**: $w \models \langle\gamma\rangle\varphi$: Angel has a **strategy** in the game γ to ensure that the game ends in a state where φ is true.

From **PDL** to Game Logic

Consequences of two players:

From **PDL** to Game Logic

Consequences of two players:

$\langle\gamma\rangle\varphi$: Angel has a strategy in γ to ensure φ is true

$[\gamma]\varphi$: Demon has a strategy in γ to ensure φ is true

From **PDL** to Game Logic

Consequences of two players:

$\langle\gamma\rangle\varphi$: Angel has a strategy in γ to ensure φ is true

$[\gamma]\varphi$: Demon has a strategy in γ to ensure φ is true

Either Angel or Demon can win: $\langle\gamma\rangle\varphi \vee [\gamma]\neg\varphi$

From **PDL** to Game Logic

Consequences of two players:

$\langle\gamma\rangle\varphi$: Angel has a strategy in γ to ensure φ is true

$[\gamma]\varphi$: Demon has a strategy in γ to ensure φ is true

Either Angel or Demon can win: $\langle\gamma\rangle\varphi \vee [\gamma]\neg\varphi$

But not both: $\neg(\langle\gamma\rangle\varphi \wedge [\gamma]\neg\varphi)$

From **PDL** to Game Logic

Consequences of two players:

$\langle\gamma\rangle\varphi$: Angel has a strategy in γ to ensure φ is true

$[\gamma]\varphi$: Demon has a strategy in γ to ensure φ is true

Either Angel or Demon can win: $\langle\gamma\rangle\varphi \vee [\gamma]\neg\varphi$

But not both: $\neg(\langle\gamma\rangle\varphi \wedge [\gamma]\neg\varphi)$

Thus, $[\gamma]\varphi \leftrightarrow \neg\langle\gamma\rangle\neg\varphi$ is a valid principle

From **PDL** to Game Logic

Consequences of two players:

$\langle\gamma\rangle\varphi$: Angel has a strategy in γ to ensure φ is true

$[\gamma]\varphi$: Demon has a strategy in γ to ensure φ is true

Either Angel or Demon can win: $\langle\gamma\rangle\varphi \vee [\gamma]\neg\varphi$

But not both: $\neg(\langle\gamma\rangle\varphi \wedge [\gamma]\neg\varphi)$

Thus, $[\gamma]\varphi \leftrightarrow \neg\langle\gamma\rangle\neg\varphi$ is a valid principle

However, $[\gamma]\varphi \wedge [\gamma]\psi \rightarrow [\gamma](\varphi \wedge \psi)$ is **not** a valid principle

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^\times := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^\times := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^\times := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^\times := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^\times := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^\times := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^\times := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $?\varphi$: Check whether φ currently holds
- $\gamma_1;\gamma_2$: First play γ_1 then γ_2
- $\gamma_1 \cup \gamma_2$: Angel choose between γ_1 and γ_2
- γ^*: Angel can choose how often to play γ (possibly not at all); each time she has played γ, she can decide whether to play it again or not.
- γ^d: Switch roles, then play γ
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between γ_1 and γ_2
- $\gamma^x := ((\gamma^d)^*)^d$: Demon can choose how often to play γ (possibly not at all); each time he has played γ, he can decide whether to play it again or not.

Game Logic

Syntax

Let Γ_0 be a set of atomic games and At a set of atomic propositions. Then formulas of Game Logic are defined inductively as follows:

$$\gamma \ := \ g \mid \varphi? \mid \gamma;\gamma \mid \gamma \cup \gamma \mid \gamma^* \mid \gamma^d$$
$$\varphi \ := \ \bot \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\gamma\rangle\varphi \mid [\gamma]\varphi$$

where $p \in \mathrm{At}, g \in \Gamma_0$.

Game Logic

A neighborhood game model is a tuple
$\mathcal{M} = \langle W, \{E_g \mid g \in \Gamma_0\}, V\rangle$ where

W is a nonempty set of states

For each $g \in \Gamma_0$, $E_g : W \to \wp(\wp(W))$ is a monotonic neighborhood function.

$X \in E_g(w)$ means in state s, Angel has a strategy to force the game to end in *some* state in X (we may write wE_gX)

$V : \mathsf{At} \to \wp(W)$ is a valuation function.

Game Logic

Propositional letters and boolean connectives are as usual.

$\mathcal{M}, w \models \langle\gamma\rangle\varphi$ iff $(\varphi)^{\mathcal{M}} \in E_\gamma(w)$

Game Logic

Propositional letters and boolean connectives are as usual.

$\mathcal{M}, w \models \langle\gamma\rangle\varphi$ iff $(\varphi)^{\mathcal{M}} \in E_\gamma(w)$

Suppose $E_\gamma(Y) := \{s \mid Y \in E_g(s)\}$

- $E_{\gamma_1;\gamma_2}(Y) \ := \ E_{\gamma_1}(E_{\gamma_2}(Y))$
- $E_{\gamma_1\cup\gamma_2}(Y) \ := \ E_{\gamma_1}(Y) \cup E_{\gamma_2}(Y)$
- $E_{\varphi?}(Y) \ := \ (\varphi)^{\mathcal{M}} \cap Y$
- $E_{\gamma^d}(Y) \ := \ \overline{E_\gamma(\overline{Y})}$
- $E_{\gamma^*}(Y) \ := \ \mu X . Y \cup E_\gamma(X)$

Game Logic: Axioms

1. All propositional tautologies
2. $\langle\alpha;\beta\rangle\varphi \leftrightarrow \langle\alpha\rangle\langle\beta\rangle\varphi$ Composition
3. $\langle\alpha \cup \beta\rangle\varphi \leftrightarrow \langle\alpha\rangle\varphi \vee \langle\beta\rangle\varphi$ Union
4. $\langle\psi?\rangle\varphi \leftrightarrow (\psi \wedge \varphi)$ Test
5. $\langle\alpha^d\rangle\varphi \leftrightarrow \neg\langle\alpha\rangle\neg\varphi$ Dual
6. $(\varphi \vee \langle\alpha\rangle\langle\alpha^*\rangle\varphi) \rightarrow \langle\alpha^*\rangle\varphi$ Mix

and the rules,

$$\frac{\varphi \qquad \varphi \rightarrow \psi}{\psi} \qquad \frac{\varphi \rightarrow \psi}{\langle\alpha\rangle\varphi \rightarrow \langle\alpha\rangle\psi} \qquad \frac{(\varphi \vee \langle\alpha\rangle\psi) \rightarrow \psi}{\langle\alpha^*\rangle\varphi \rightarrow \psi}$$

Game Logic

- Game Logic is more expressive than **PDL**

Game Logic

- Game Logic is more expressive than **PDL**

$$\langle (g^d)^* \rangle \bot$$

Game Logic

- Game Logic is more expressive than **PDL**

$$\langle (g^d)^* \rangle \bot$$

- The induction axiom is not valid in GL.

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

Game Logic

- Game Logic is more expressive than **PDL**

$$\langle (g^d)^* \rangle \bot$$

- The induction axiom is not valid in GL.

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

- All GL games are determined.

Game Logic

Theorem Dual-free game logic is sound and complete with respect to the class of all game models.

Game Logic

Theorem Dual-free game logic is sound and complete with respect to the class of all game models.

Theorem Iteration-free game logic is sound and complete with respect to the class of all game models.

Game Logic

Theorem Dual-free game logic is sound and complete with respect to the class of all game models.

Theorem Iteration-free game logic is sound and complete with respect to the class of all game models.

Open Question Is (full) game logic complete with respect to the class of all game models?

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

M. Pauly. *Logic for Social Software*. Ph.D. Thesis, University of Amsterdam (2001)..

Game Logic

Theorem Given a game logic formula φ and a finite game model $\mathcal{M}$, model checking can be done in time $O(|\mathcal{M}|^{ad(\varphi)+1} \times |\varphi|)$

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

M. Pauly. *Logic for Social Software*. Ph.D. Thesis, University of Amsterdam (2001)..

D. Berwanger. *Game Logic is Strong Enough for Parity Games*. Studia Logica **75** (2003)..

Game Logic

Theorem The satisfiability problem for game logic is in EXPTIME.

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

M. Pauly. *Logic for Social Software*. Ph.D. Thesis, University of Amsterdam (2001)..

D. Berwanger. *Game Logic is Strong Enough for Parity Games*. Studia Logica **75** (2003)..

Game Logic

Theorem Game logic can be translated into the modal μ-calculus

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

M. Pauly. *Logic for Social Software*. Ph.D. Thesis, University of Amsterdam (2001)..

D. Berwanger. *Game Logic is Strong Enough for Parity Games*. Studia Logica **75** (2003)..

Game Logic

Theorem Game logic can be translated into the modal μ-calculus

Theorem No finite level of the modal μ-calculus hierarchy captures the expressive power of game logic.

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

M. Pauly. *Logic for Social Software*. Ph.D. Thesis, University of Amsterdam (2001)..

D. Berwanger. *Game Logic is Strong Enough for Parity Games*. Studia Logica **75** (2003)..

Game Algebra

Definition Two games γ_1 and γ_2 are equivalent provided $E_{\gamma_1} = E_{\gamma_2}$ in all models

Game Algebra

Definition Two games γ_1 and γ_2 are equivalent provided $E_{\gamma_1} = E_{\gamma_2}$ in all models (iff $\langle\gamma_1\rangle p \leftrightarrow \langle\gamma_2\rangle p$ is valid for a p which occurs neither in γ_1 nor in γ_2.)

Game Algebra

Game Boards: Given a set of states or positions B, for each game g and each player i there is an associated relation $E_g^i \subseteq B \times 2^B$:

$pE_g^i T$ holds if in position p, i can force that the outcome of g will be a position in T.

- (monotonicity) if $pE_g^i T$ and $T \subseteq U$ then $pE_g^i U$
- (consistency) if $pE_g^i T$ then not $pE_g^{1-i}(B - T)$

Given a game board (a set B with relations E_g^i for each game and player), we say that two games g, h ($g \approx h$) are equivalent if $E_g^i = E_h^i$ for each i.

Game Algebra

Game Algebra

1. Standard Laws of Boolean Algebras

Game Algebra

1. Standard Laws of Boolean Algebras
2. $(x; y); z \approx x; (y; z)$

Game Algebra

1. Standard Laws of Boolean Algebras
2. $(x; y); z \approx x; (y; z)$
3. $(x \vee y); z \approx (x; z) \vee (y; z)$, $(x \wedge y); z \approx (x; z) \wedge (y; z)$

Game Algebra

1. Standard Laws of Boolean Algebras
2. $(x; y); z \approx x; (y; z)$
3. $(x \vee y); z \approx (x; z) \vee (y; z)$, $(x \wedge y); z \approx (x; z) \wedge (y; z)$
4. $-x; -y \approx -(x; y)$

Game Algebra

1. Standard Laws of Boolean Algebras
2. $(x; y); z \approx x; (y; z)$
3. $(x \vee y); z \approx (x; z) \vee (y; z)$, $(x \wedge y); z \approx (x; z) \wedge (y; z)$
4. $-x; -y \approx -(x; y)$
5. $y \preceq z \Rightarrow x; y \preceq x; z$

Theorem Sound and complete axiomatizations of (iteration free) game algebra

Y. Venema. *Representing Game Algebras*. Studia Logica **75** (2003)..

V. Goranko. *The Basic Algebra of Game Equivalences*. Studia Logica **75** (2003)..

Concurrent Game Logic

$\gamma_1 \cap \gamma_2$ means "play γ_1 and γ_2 in parallel."

Concurrent Game Logic

$\gamma_1 \cap \gamma_2$ means "play γ_1 and γ_2 in parallel."

Need both the disjunctive and conjunctive interpretation of the neighborhoods.

Main Idea: $R_\gamma \subseteq W \times \wp(\wp(\wp(W)))$

J. van Benthem, S. Ghosh and F. Liu. *Modelling Simultaneous Games in Dynamic Logic*. LORI (2007).

More Information on Game Logic and Algebra

Editors: M. Pauly and R. Parikh. *Special Issue on Game Logic*. Studia Logica **75**, 2003.

M. Pauly and R. Parikh. *Game Logic — An Overview*. Studia Logica **75**, 2003.

R. Parikh. *The Logic of Games and its Applications.*. Annals of Discrete Mathematics. (1985) .

J. van Benthem. *Notes on Logics and Games*. 2007.

Thank You!