FIL2405/4405: Simple Quantified Modal Logic

3rd March

Introduction

So far, we have looked at propositional modal logic.

This week, we extend our study of modal logic to include quantifiers.

We will look at some simple first-order modal logics.

Lower Predicate Calculus

Let's look at non-modal first-order logic, or Lower Predicate Calculus.

Lower Predicate Calculus allows us to investigate inferences and logical relations which propositional logic is not strong enough to capture.

Consider the following valid argument.

(1) There are at least two cats.
$\therefore$ (2) There is at least one cat.

Using propositional logic, we can at best say that the argument (1)–(2) has the form $p \therefore q$. This is an invalid form. But (1)–(2) is valid.

Lower Predicate Calculus

The language of Lower Predicate Calculus ($\mathcal{L}_\forall$) is more expressive than $\mathcal{L}_\rho$ and allows us to investigate the logical relations between parts of sentences.

The innovation: introduce predicates, variables and quantifiers.

Think of a predicate as a certain condition which things can satisfy.

Sally is a cat

Lower Predicate Calculus

The language of Lower Predicate Calculus ($\mathcal{L}_\forall$) is more expressive than $\mathcal{L}_\rho$ and allows us to investigate the logical relations between parts of sentences.

The innovation: introduce predicates, variables and quantifiers.

Think of a predicate as a certain condition which things can satisfy.

Sally is a cat

Lower Predicate Calculus

The language of Lower Predicate Calculus ($\mathcal{L}_\forall$) is more expressive than $\mathcal{L}_\rho$ and allows us to investigate the logical relations between parts of sentences.

The innovation: introduce predicates, variables and quantifiers.

Think of a predicate as a certain condition which things can satisfy.

Sally is a cat
Sally is on the table

Lower Predicate Calculus

The language of Lower Predicate Calculus ($\mathcal{L}_\forall$) is more expressive than $\mathcal{L}_\rho$ and allows us to investigate the logical relations between parts of sentences.

The innovation: introduce predicates, variables and quantifiers.

Think of a predicate as a certain condition which things can satisfy.

Sally is a cat
Sally is on the table

Here '... is a cat' is a one-place predicate and '... is on ...' is a two-place predicate. One thing satisfies the first, two things satisfy the second.

In $\mathcal{L}_\forall$ we use greek letters for predicates, i.e., $\phi, \psi, \chi, \ldots$

Lower Predicate Calculus

Think of variables as place-holders for where a name could go.

For instance, just like 'Sally is a cat', we could write 'x is a cat'.

Variables combine with quantifiers. There are two:

(∃) '∃' for 'there exists ...', or 'there is ...', or 'there is at least one ...'.
(∀) '∀' for 'for every...', or 'for any...', or 'every ...'.

For instance, read '$\exists x(x$ is a cat)' as 'there is at least one cat'.

For instance, read '$\forall x(x$ is a cat)' as 'Everything is a cat'.

Lower Predicate Calculus

Here's the precise definition of $\mathcal{L}_\forall$, the lexicon and grammar.

The Lexicon of the Language of Lower Predicate Calculus $\mathcal{L}_\forall$

For each natural number n $(n \geq 1)$, we have denumerably many n-place predicates, ϕ^n, ψ^n, χ^n. We have denumerably many individual variables, x, y, z. We have the logical symbols: $\sim, \wedge, \vee, \supset, \equiv, \forall, \exists$. Finally, we have, as punctuation, bracket symbols (, and).

For convenience, we usually write ϕ^n, ψ^n, χ^n as ϕ, ψ, χ

All of $\sim, \wedge, \vee, \supset, \equiv, \forall, \exists$ are primitive. Though, they don't have to be:

Sometimes, $\exists x \phi x =_{df} \sim\forall x \sim\phi x$

Lower Predicate Calculus

Grammar of the Language of First Order Logic $\mathcal{L}_\forall$

For any n-place predicate, ϕ^n, and n many variables, $x_1, ..., x_n$, $\phi x_1, ..., x_n$ is an *atomic wff*. Moreover:

(i) If α is a *wff*, then $\sim\alpha$ is a *wff*.

(ii) If α and β are *wffs*, then $\alpha \wedge \beta$, $\alpha \vee \beta$, $\alpha \supset \beta$, and $\alpha \equiv \beta$ are *wffs*.

(iii) If α is a *wff* and x a variable, then $\forall x\alpha$ and $\exists x\alpha$ are *wffs*.

With (iii) we have to be careful to not introduce ambiguity.

If $\alpha = \phi x \wedge \psi x$, then $\forall x\alpha = \forall x(\phi x \wedge \psi x)$.

If $\alpha = \phi x$, just write $\forall x\alpha = \forall x\phi x$.

Lower Predicate Calculus Semantics

Quantifiers are associated with domains, i.e., collections of things.

Take a *wff* of $\mathcal{L}_\forall$, e.g., $\forall x \exists y \phi xy$. We can *interpret* this:

- Let ϕ be the predicate 'x hates y'.
- Let the quantifiers range over all people.
- '$\forall x \exists y \phi xy$' means 'Everyone hates someone'.

(Alternatively, let the quantifiers range over cats:
'$\forall x \exists y \phi xy$' means 'Every cat hates some cat'.)

The formal semantics of $\mathcal{L}_\forall$ generalises this interpreting.

Lower Predicate Calculus Semantics

We want our semantics to be *compositional*.

Just like in PL, we use valuation functions v to interpret *wff* of $\mathcal{L}_\forall$.

Unlike in PL, v also interprets sub-sentential expressions, e.g,

v interprets predicates, e.g., ϕ, ψ, χ, ...

To interpret $\mathcal{L}_\forall$, we need a domain. The interpretations (or models) of $\mathcal{L}_\forall$ are pairs $\langle D, v\rangle$, where D is a non-empty set and v is a valuation.

Lower Predicate Calculus Semantics (Predicates)

v interprets the predicates by assigning *extensions*.

Intuitively, e.g., the extension of '... is blue' is the set of all blues things.

The extensions of one-place predicates, given D, is just a subset of D.

But n-place predicates, where $n > 1$ are more complicated. We want to preserve the order in which the elements are related. Generalising:

Given D, $v(\phi^n)$ is a set of n-tuples $\langle u_1, ..., u_n \rangle$, $u_1, ..., u_n \in D$.

E.g., if ϕxy is x loves y, then $v(\phi)$ is a set of $\langle u_1, u_2 \rangle$ where u_1 loves u_2.

Lower Predicate Calculus Semantics (Variables)

We want to determine, given an interpretation, whether a *wff* is true.

Some *wff* of $\mathcal{L}_\forall$ contain free variables. A free variable is not bound by any quantifier, e.g., variables in bold are free:

$$\phi\mathbf{x},\ \forall x\phi\mathbf{y}x,\ \forall y\forall x((\phi x \wedge \psi xy) \rightarrow \phi\mathbf{z})$$

Is 'ϕx' true? It depends on what *x* is! For this, we use assignments.

Value Assignments μ

Where $\langle D, v\rangle$ is a model, we say that μ is a value-assignment based on $\langle D, v\rangle$ provided that, for every variable x in $\mathcal{L}_\forall$, $\mu(x) \in D$. We write $v_\mu(\alpha) = 1$ if α is true in the model $\langle D, v\rangle$, given the assignment μ.

Lower Predicate Calculus Semantics (Variables)

We want $\forall x \phi x$ to be true in a model if ϕx is true for any value of x.

To capture this, we use the notion of an x-alternative of μ.

An x-alternative of μ.

If μ is a value assignment, let ρ be the x-alternative of μ iff for every variable y except (possibly) x, $\rho(y) = \mu(y)$.

$$\mu \begin{cases} \mu(x) = 1 \\ \mu(y) = 2 \\ \mu(z) = 3 \end{cases} \qquad \mu' \begin{cases} \mu'(x) = 2 \\ \mu'(y) = 2 \\ \mu'(z) = 3 \end{cases}$$

Lower Predicate Calculus Semantics (Variables)

We want $\forall x \phi x$ to be true in a model if ϕx is true for any value of x.

To capture this, we use the notion of an x-alternative of μ.

An x-alternative of μ.

If μ is a value assignment, let ρ be the x-alternative of μ iff for every variable y except (possibly) x, $\rho(y) = \mu(y)$.

$$\mu \begin{cases} \mu(x) = 1 \\ \mu(y) = 2 \\ \mu(z) = 3 \end{cases} \qquad \mu' \begin{cases} \mu'(x) = 2 \\ \mu'(y) = 2 \\ \mu'(z) = 3 \checkmark \end{cases}$$

Lower Predicate Calculus Semantics (Variables)

We want $\forall x \phi x$ to be true in a model if ϕx is true for any value of x.

To capture this, we use the notion of an x-alternative of μ.

An x-alternative of μ.

If μ is a value assignment, let ρ be the x-alternative of μ iff for every variable y except (possibly) x, $\rho(y) = \mu(y)$.

$$\mu \begin{cases} \mu(x) = 1 \\ \mu(y) = 2 \\ \mu(z) = 3 \end{cases} \qquad \mu' \begin{cases} \mu'(x) = 2 \\ \mu'(y) = 2 \\ \mu'(z) = 3 \checkmark \end{cases} \qquad \mu'' \begin{cases} \mu''(x) = 1 \\ \mu''(y) = 3 \\ \mu''(z) = 3 \end{cases}$$

Lower Predicate Calculus Semantics (Variables)

We want $\forall x \phi x$ to be true in a model if ϕx is true for any value of x.

To capture this, we use the notion of an x-alternative of μ.

An x-alternative of μ.

If μ is a value assignment, let ρ be the x-alternative of μ iff for every variable y except (possibly) x, $\rho(y) = \mu(y)$.

$$\mu \begin{cases} \mu(x) = 1 \\ \mu(y) = 2 \\ \mu(z) = 3 \end{cases} \qquad \mu' \begin{cases} \mu'(x) = 2 \\ \mu'(y) = 2 \\ \mu'(z) = 3 \checkmark \end{cases} \qquad \mu'' \begin{cases} \mu''(x) = 1 \\ \mu''(y) = 3 \\ \mu''(z) = 3 \times \end{cases}$$

Lower Predicate Calculus Semantics (Variables)

We want $\forall x \phi x$ to be true in a model if ϕx is true for any value of x.

To capture this, we use the notion of an x-alternative of μ.

An x-alternative of μ.

If μ is a value assignment, let ρ be the x-alternative of μ iff for every variable y except (possibly) x, $\rho(y) = \mu(y)$.

$$\mu \begin{cases} \mu(x) = 1 \\ \mu(y) = 2 \\ \mu(z) = 3 \end{cases} \quad \mu' \begin{cases} \mu'(x) = 2 \\ \mu'(y) = 2 \\ \mu'(z) = 3 \checkmark \end{cases} \quad \mu'' \begin{cases} \mu''(x) = 1 \\ \mu''(y) = 3 \\ \mu''(z) = 3 \times \end{cases} \quad \mu^* \begin{cases} \mu^*(x) = 1 \\ \mu^*(y) = 2 \\ \mu^*(z) = 3 \end{cases}$$

Lower Predicate Calculus Semantics (Variables)

We want $\forall x \phi x$ to be true in a model if ϕx is true for any value of x.

To capture this, we use the notion of an x-alternative of μ.

An x-alternative of μ.

If μ is a value assignment, let ρ be the x-alternative of μ iff for every variable y except (possibly) x, $\rho(y) = \mu(y)$.

$$\mu \begin{cases} \mu(x) = 1 \\ \mu(y) = 2 \\ \mu(z) = 3 \end{cases} \quad \mu' \begin{cases} \mu'(x) = 2 \\ \mu'(y) = 2 \\ \mu'(z) = 3 \checkmark \end{cases} \quad \mu'' \begin{cases} \mu''(x) = 1 \\ \mu''(y) = 3 \\ \mu''(z) = 3 \times \end{cases} \quad \mu^* \begin{cases} \mu^*(x) = 1 \\ \mu^*(y) = 2 \\ \mu^*(z) = 3 \checkmark \end{cases}$$

Lower Predicate Calculus Semantics

With all this, we can now define truth in a model for *wff* of $\mathcal{L}_\forall$.

Truth in a model $\langle D, v \rangle$ under assignment μ

If $\langle D, v \rangle$ is a model, where D is a non-empty set and v some valuation function, α and β are *wff* of $\mathcal{L}_\forall$, and $x_1, \ldots, x_n$ are variables, then, given some value assignment μ:

(ϕ^v) $v_\mu(\phi x_1, \ldots, x_n) = 1$ if $\langle \mu(x_1), \ldots, \mu(x_n) \rangle \in v(\phi)$; 0 otherwise.

($\sim^v$) $v_\mu({\sim}\alpha) = 1$ if $v_\mu(\alpha) = 0$; 0 otherwise.

($\wedge^v$) $v_\mu(\alpha \wedge \beta) = 1$ if $v_\mu(\alpha) = 1$ and $v_\mu(\beta) = 1$; 0 otherwise.

… *and so on for the logical connectives* …

($\forall^v$) $v_\mu(\forall x \alpha) = 1$ if $v_\rho(\alpha) = 1$, for any x-alternative ρ of μ; 0 otherwise.

($\exists^v$) $v_\mu(\exists x \alpha) = 1$ if $v_\rho(\alpha) = 1$, for some x-alternative ρ of μ; 0 otherwise.

First-Order Validity.

Valid in $\langle D, v\rangle$

A *wff* α of $\mathcal{L}_\forall$ is valid in a model $\langle D, v\rangle$ iff $v_\mu(\alpha) = 1$ for every assignment μ in $\langle D, v\rangle$ to the variables of $\mathcal{L}_\forall$.

Valid simpliciter

A *wff* α of $\mathcal{L}_\forall$ is valid simpliciter if it is valid in every model $\langle D, v\rangle$.

Axiomatizing Lower Predicate Calculus

To characterise LPC syntactically, we need to be precise about two things.

Replacing variables in *wff*

We will often write $\alpha[y/x]$, 'α, replacing x for y'. This is the result of replacing every free x in α with a y such that y is also free.

For example, $\alpha := Rxyz \Rightarrow \alpha[x/t] := Rtyz \Rightarrow \alpha[x/y] := Ryyz$

LPC Substitution-Instances

An LPC Substitution-Instance of a *wff* of Propositional Logic is an expression which results from uniformly replacing every propositional variable in α by a *wff* of $\mathcal{L}_\forall$.

For example, $p \supset p \Rightarrow Fx \supset Fx$. (Replace p with Fx)

LPC Axiomatized

The axioms of LPC, where α and β are *wff* of $\mathcal{L}_\forall$

PC Any LPC substitution-instance of a valid *wff* of PC is an axiom.

$\forall$1 If x and y any variables then $\forall x\alpha \supset \alpha[y/x]$ is an axiom.

The transformation rules:

MP If $\vdash \alpha$, $\vdash \alpha \supset \beta$, then $\vdash \beta$.

$\forall$2 If $\vdash \alpha \supset \beta$, then $\alpha \supset \forall x\beta$, provided x is not free in α.

Language of Modal LPC

The language of Modal LPC, $\mathcal{L}_\forall^M$ is simply $\mathcal{L}_\forall$ extended to handle L.

Lexicon of $\mathcal{L}_\forall^M$

The lexicon of $\mathcal{L}_\forall^M$ is the lexicon of $\mathcal{L}_\forall$ extended to include L.

Grammar of $\mathcal{L}_\forall^M$

The grammar of $\mathcal{L}_\forall^M$ is the grammar of $\mathcal{L}_\forall$ extended to include:

(L) If α is a *wff* of $\mathcal{L}_\forall^M$, then $L\alpha$ is a *wff* of $\mathcal{L}_\forall^M$.

Systems of Modal LPC

We can define Modal LPC correlates of the propositional modal logic.

Definition of System LPC + S

Let S be a system of normal propositional modal logic. The axioms and inference rules of LPC + S are as follows. Three axioms:

(S′) Any LPC substitution-instance of a theorem of S is an axiom.

(∀1) If α is any *wff* and x, y variables, then $\forall x\alpha \supset \alpha[y/x]$ is an axiom.

(BF) If $\alpha \in \mathcal{L}_\forall^M$, then $\forall x L\alpha \rightarrow L\forall x\alpha$ is a theorem of LPC + S.

Three inference rules:

(N) If α is a theorem, then $L\alpha$ is a theorem.

(MP) If α is a theorem and $\alpha \supset \beta$ is a theorem, then β is a theorem.

(∀2) If $\alpha \supset \beta$ is a theorem and x is not free in α, $\alpha \supset \forall x\beta$ is a theorem.

Systems of Modal LPC

LPC + K is the system which contains all the LPC substitution-instances of theorems of K as axioms. For instance:

$L(\forall x\phi x \supset \exists x\phi x) \supset (L\forall x\phi x \supset L\exists x\phi x)$ is an axiom of LPC + K

LPC + S4 is the system which contains all the LPC substitution-instances of theorems of S4 as axioms. For instance:

$L\forall x\forall y\psi xy \supset LL\forall x\forall y\psi xy$ is an axiom of LPC + S4.

LPC + 5 is often known as SQML 'Simple Quantified Modal Logic'.

Semantics for Modal LPC

Extend the notion of a model for Lower Predicate Calculus:

(i) We include a set of worlds W and accessbility relation R.

(ii) v assigns each predicate a set of $n+1$ tuples, including a $w \in W$.

Modal LPC Model

A model for Modal LPC $\langle W, R, D, v \rangle$ is a 4-tuple, where W is a non-empty set, R is a binary relation on W, D is a non-empty set, and v is a valuation function such that v assigns, for every n-place predicate ϕ in $\mathcal{L}_\forall^M$, a set of $n+1$ tuples $\langle u_1, ..., u_n, w \rangle$, for each $w \in W$.

Semantics of $\mathcal{L}_\forall^M$

Truth in an Modal LPC model at a world is given as follows.

Truth in an Modal LPC Model

Let μ be an assignment to the variables such that for each variable x, $\mu(x) \in D$. Then, every *wff* has a truth-value at a world in the model, under an assignment, as determined by the following:

(ϕ^v) $v_\mu(\phi x_1 ... x_n, w) = 1$ if $\langle \mu(x_1), ..., \mu(x_n), w \rangle \in v(\phi)$; 0 otherwise.

($\sim^v$) $v_\mu({\sim}\alpha, w) = 1$ if $v_\mu(\alpha, w) = 0$; 0 otherwise.

... *and so on for the other logical connectives* ...

($\forall^v$) $v_\mu(\forall x\alpha, w) = 1$ if $v_\rho(\alpha, w) = 1$, for every x-alternative ρ of μ; 0 otherwise.

($\exists^v$) $v_\mu(\exists x\alpha, w) = 1$ if $v_\rho(\alpha, w) = 1$, for some x-alternative ρ of μ; 0 otherwise.

(L^v) $v_\mu(L\alpha, w) = 1$ if $v_\mu(\alpha, w') = 1$ for every w' such that Rww'; 0 otherwise.

Examples

Consider $\mathfrak{M} = \langle W, R, D, v \rangle$, where $W = \{w_1, w_2\}$, $R : Rw_1w_1$, Rw_1w_2 and Rw_2w_1, $D = \{1, 2\}$, $v(\phi^1) = \{\langle 1, w_1 \rangle, \langle 2, w_1 \rangle, \langle 2, w_2 \rangle\}$

1. $\mathfrak{M}, w_1, \mu \vDash \phi x$, where $\mu(x) = 1$?
2. $\mathfrak{M}, w_2, \mu \vDash \phi x$, where $\mu(x) = 1$?
3. $\mathfrak{M} \vDash \forall x \phi x$?
4. $\mathfrak{M}, w_2 \vDash L\forall x \phi x$?
5. $\mathfrak{M}, w_1 \vDash L\forall x \phi x$?

Soundness

Modal LPC models are based on the same frames as models for propositional modal logic. So we can extend previous results.

Soundness for Modal LPC

Each of the following systems of Modal LPC is sound with respect to the class of frames listed beside it.

- LPC + K : all frames
- LPC + T : reflexive frames
- LPC + B : reflexive and symmetric frames
- LPC + S4 : reflxive and transitive frames
- LPC + S5 (SQML) : equivalence frames

For instance, if $\mathfrak{M}^R$ is an arbitrary reflexive model based on an arbitrary reflexive frames, then $\vdash_{\text{LPC+T}} \alpha$, then $\mathfrak{M}^R \vDash \alpha$.

Completeness

For our purposes, we also have completeness results.

Completeness for Modal LPC

Each of the following systems of Modal LPC is complete with respect to the class of frames listed beside it.

- LPC + K : all frames
- LPC + T : reflexive frames
- LPC + B : reflexive and symmetric frames
- LPC + S4 : reflxive and transitive frames
- LPC + S5 (SQML) : equivalence frames

For instance, if $\mathfrak{M}^{\mathsf{RT}}$ is an arbitrary model based on an arbitrary reflexive and transitive frame, then if $\mathfrak{M}^{\mathsf{RT}} \vDash \alpha$, then $\vdash_{\mathsf{LPC+S4}} \alpha$.

Summary

- We looked first-order and first-order modal languages.
- We looked at the semantics and proof theory for first-order non-modal logic, i.e., the Lower Predicate Calculus
- We looked at the semantics and proof theory for first-order modal logics, i.e., Modal Lower Predicate Calculus.
- We saw how to semantically and syntactically define Lower Predicate Calculus correlates of K, T, B, S4, and S5.