Worksheet 6

First Order Predicate Calculus (Recap)

In this worksheet we revise and consolidate material covered in the TOC course. The aims are:

- to review the syntax, semantics (model theory) and one proof theory for FOPC;
- to show that the proof theory is not easily automated (thus motivating the alternative approach taken in the next worksheets).

6.1 The Syntax of First-Order Predicate Calculus

Syntax determines how legal statements called *well-formed formulas* (wffs) may be constructed.

Terms are defined recursively as follows:

- A constant is a term.
- A variable is a term.
- If f is a n-place function symbol, and $t_1,\ldots,t_n$ are terms, then $f(t_1,\ldots,t_n)$ is a term.
- All terms are generated by applying the above rules.

Atoms (or atomic formulas) are defined as follows:

- If $t_1, \ldots, t_n$ are terms and P is a n-ary predicate symbol then $P(t_1, \ldots, t_n)$ is an atom. No other expression is an atom.

Well-formed formulas, wffs, are defined recursively as follows:

- Any atom is a wff.
- If α and β are wffs, and V is a variable, then the following are also wffs (shown here with their names and informal meanings):

wff	name	informal meaning
$\neg\alpha$	Negation	α is false
$(\alpha \rightarrow \beta)$	Material implication	If α then β
$(\alpha \vee \beta)$	Disjunction	Either α or β (or both)
$(\alpha \wedge \beta)$	Conjunction	Both α and β
$(\alpha \leftrightarrow \beta)$	Material Equivalence	α if and only if β
$\forall V \alpha$	Universal Quantification	For all V, α
$\exists V \alpha$	Existential Quantification	There is at least one V such that α

- Wffs are generated only by a finite number of applications of the above rules.

Notes

- Predicate symbols, function symbols, variables and constants are drawn from disjoint sets. As a convention, we will use lowercase letters and strings for predicate symbols (e.g. p, q, elephant, mortal), lowercase letters such as f, g and h, subscripted as necessary, for function symbols, uppercase letters from the end of the alphabet (X, Y, Z), subscripted as necessary, for variables, and lowercase letters from the beginning of the alphabet and strings (a, b, c, clyde, chas, di) for constants.
- A constant may be thought of as a function that takes no arguments.
- You might be wondering what the difference between a function and a predicate is. There is obviously a syntactic difference: a function and its arguments can build terms; predicates cannot. Then, of course, there is a semantic difference which we will come to later, but which I will prefigure here: roughly, a function takes one or more entities and maps them to an entity, e.g. author-of(the-spire) represents an entity (e.g. in our world *William Golding*); a predicate takes one or more entities and maps them to a truth-value, e.g. author(the-spire, william-golding) is (in our world) true. You can see that they are quite different, but also that we might have a choice about how to represent something, i.e. whether as a function or as a predicate.[1]

[1] We have seen that functions turn out to be useful in logic programming for defining complex data structures, such as lists.

- It is possible for a predicate to be of arity 0 (no arguments). In this case, it is sometimes referred to as a proposition symbol (c.f. propositional logic in TOC).
- You might also see the symbol $\supset$ in place of $\rightarrow$, and $\equiv$ in place of $\leftrightarrow$.
- $\alpha \vee \beta$ is a *disjunction*; α and β are the *disjuncts* of $\alpha \vee \beta$.
- $\alpha \wedge \beta$ is a *conjunction*; α and β are *conjuncts* of $\alpha \wedge \beta$.
- The typical precedence of the connectives and quantifiers from highest to lowest is:

  $$\neg \vee \wedge \rightarrow \leftrightarrow \forall \exists$$

  We can use this to omit parentheses. (If you're ever unsure, use lots of parentheses!) For example,

  $$\exists X\ \mathrm{p}(X) \vee \mathrm{q} \rightarrow \forall Y\ \mathrm{r}(X, Y)$$

  stands for

  $$\exists X\ ((\mathrm{p}(X) \vee \mathrm{q}) \rightarrow \forall Y\ \mathrm{r}(X, Y)).$$

  If you did not want the existential to extend to the second disjunct, then different brackets would be needed, i.e.

  $$(\exists X\ \mathrm{p}(X)) \vee \mathrm{q} \rightarrow \forall Y\ \mathrm{r}(X, Y).$$

  which stands for

  $$((\exists X\ \mathrm{p}(X)) \vee \mathrm{q}) \rightarrow \forall Y\ \mathrm{r}(X, Y).$$

- The *scope* of a quantifier occurring in a formula is the subformula to which the quantifier applies. Re-using the previous example, the scope of existential quantification of X in

  $$\exists X\ \mathrm{p}(X) \vee \mathrm{q} \rightarrow \forall Y\ \mathrm{r}(X, Y)$$

  is

  $$\mathrm{p}(X) \vee \mathrm{q} \rightarrow \forall Y\ \mathrm{r}(X, Y));$$

  its scope in

  $$(\exists X\ \mathrm{p}(X)) \vee \mathrm{q} \rightarrow \forall Y\ \mathrm{r}(X, Y).$$

  is $\mathrm{p}(X)$. The scope of the universal quantification of Y is $\mathrm{r}(X, Y)$ in both cases.

- A variable X in a formula α is *bound* in α if it occurs within a subformula of the form $\forall X \beta$ or $\exists X \beta$, i.e. X is bound if it occurs within the scope of a quantifier that employs X. Otherwise, X is *free* in α. For example, in

$$\exists X \ \mathrm{p}(X) \vee \mathrm{q} \rightarrow \forall Y \ \mathrm{r}(X, Y)$$

X and Y are both bound; in

$$\forall X \ \mathrm{p}(X, Y)$$

X is bound but Y is free; in

$$(\exists X \ \mathrm{p}(X)) \vee \mathrm{q} \rightarrow \forall Y \ \mathrm{r}(X, Y).$$

Y is bound and X is *both* free and bound (the first occurrence, where it is an argument to p, is bound, but the second occurrence, where it is an argument to r is not within the scope of a quantifier on X and so is free).

- In this course, we shall stipulate that we only deal in wffs in which all variables are bound (no free variables). Wffs in which all variables are bound are sometimes referred to as *sentences*. However, I will tend to continue to use the phrase 'wff'; when I do, I really mean 'sentence'.

- Finally, note that *first order* logic does *not* allow predicate or function variables, and it does *not* allow predicates applied to predicates. This is what makes it *first-order*. For example, the following are *not* permitted:

  $$\forall p \forall X (p(X) \rightarrow \mathrm{q}(X))$$
  $$\forall X \ \mathrm{q}(\mathrm{q}(X))$$

  For these, we would need a *higher order logic*.

Of course, all this stuff about the syntax of the logic is only part of the problem. The real problem when writing knowledge based systems is choosing the 'right' predicate symbols, etc. For example, to represent "The Sun is yellow" do we write: yellow(sun), colour(sun, yellow) or property(sun, colour, yellow)? The first is good if we often need to answer questions about which knowledge based objects are yellow, the second is good if we often need to answer questions about the colours of objects, and the third is good if we often need to answer questions about what properties objects have.

6.2 The Semantics of First Order Predicate Calculus

6.2.1 Informal Presentation

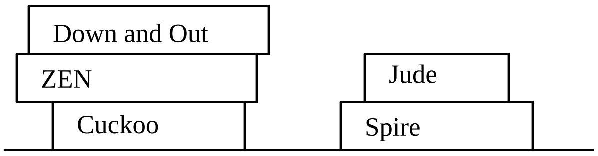
Semantics ascribes meaning to wffs. This is done by ascribing meaning to the parts of a wff and combining these meanings appropriately.

Ascription of meaning to a wff is going to boil down to knowing when the wff is true, i.e. knowing which states of affairs the wffs are true in. For this, we have the notion of a *world*. A world consists of a set of entities or objects, U, and interrelationships defined on U.

The set of entities is sometimes referred to as the *domain of discourse* or *universe of discourse*. There can be any number of entities, possibly an infinite number. The notion of an entity is quite broad. Entities can be concrete (e.g. the lake, this piece of paper) or abstract (e.g. the number 2, the set of all integers, the concept of justice). Entities can be fictional or imaginary (e.g. Winnie the Pooh, a unicorn). In short, an entity can be any object of thought.

A simple example is the world shown in the picture:

The Books World



There are five entities (excluding the table), each one being a book.

There are two kinds of interrelationship. First, there are *functions*. These map tuples of entities into other entities. For example, in the Books World we have a function that maps from a book to the book directly on top of it (this is a partial function since some books have nothing on top of them).

Secondly, there are *relations*. A relation is a set of tuples of entities. In the Books World, we have relations such as the one telling us whether a pair of books is such that the second is somewhere above the first. Another relation tells us whether a book has a clear top, and so on.

We are going to define meaning relative to a world. But if we are to be able to do this, we need a way of relating the symbols we use in our wffs to the things of our world. We do this using the notion of an *interpretation*, I.

An interpretation is a function which maps symbols to things in the world. Of course, for a particular set of symbols and for a particular world, there are many possible interpretation functions.

An interpretation function, I, makes the following mappings:

1. For every constant c, I(c) is an entity from U.[2]

2. For every n-place function symbol f, I(f) is an n-place function in the world, which maps tuples of n entities onto an entity.

[2] Note that in the semantics of FOPC constants need not denote different entities (i.e. an entity in the world can have two names), and there may be unnamed entities. Likewise for function and predicate symbols.

3. For every n-ary predicate symbol p, $\mathrm{I}(p)$ is an n-ary relation in the world, which maps every tuple of n entities to true or false.

The difference between a predicate symbol and a function symbol should now be clear. A predicate with its arguments makes a statement which is true or false w.r.t an interpretation. A function symbol with its arguments does not make a statement. Rather, under an interpretation, it denotes an entity just like a constant does.

Note that since the interpretation function fixes the denotation of a constant, function or predicate symbol, the apparent English meaning of the constant, function or predicate symbol is strictly irrelevant. A constant kanga could be made to denote the world entity *Tigger* in one interpretation, or *Piglet* in another. A predicate loves could be made to denote the world relationship *hates* in one interpretation, or *less than* in another. Of course, convenience suggests that we should not be so awkward: the constant or predicate name should be a mnemonic for the denotation that is most likely.

For example, for our Books World given earlier, and for a set of constants a, b, c, d, and e, a function symbol f and two predicate symbols p and q, one interpretation, I_1, might be:

I_1(a)	=	Down and Out in London and Paris
I_1(b)	=	ZEN and the Art of Motorcycle Maintenance
I_1(c)	=	One Flew Over the Cuckoo's Nest
I_1(d)	=	Jude the Obscure
I_1(e)	=	The Spire
I_1(f)	=	the function that maps from a book to the book directly on top of it, i.e. {(ZEN $\mapsto$ Down and Out), (Cuckoo $\mapsto$ ZEN), (Spire $\mapsto$ Jude)}.
I_1(p)	=	the relation satisfied by pairs of books where the second is directly on top of the first, i.e. {<ZEN, Down and Out>, <Cuckoo, ZEN>, <Spire, Jude>}.
I_1(q)	=	the relation satisfied by pairs of books where the second is somewhere above the first, i.e. {<ZEN, Down and Out>, <Cuckoo, ZEN>, <Cuckoo, Down and Out> <Spire, Jude>}.

But we could just as well have an interpretation, I_2, which is identical to I_1 except that:

I_2(a) = ZEN and the Art of Motorcycle Maintenance
I_2(b) = Down and Out in London and Paris
I_2(p) = the relation satisfied by pairs of books where the second is directly below the first, i.e. {<Down and Out, ZEN>, <ZEN, Cuckoo>, <Jude, Spire>}.
I_2(q) = the relation satisfied by pairs of books where the second is directly on top of the first, i.e. {<ZEN, Down and Out>, <Cuckoo, ZEN>, <Spire, Jude>}.

. . . or any number of other interpretations.

We can now use the interpretation function to determine the truth value of a wff. We begin with an informal presentation.

For a given interpretation, I, and universe of discourse, U, a wff can be evaluated to true or false according to the following rules:

• A simple atom $p(t_1,\ldots,t_n)$ for predicate symbol p and terms $t_1,\ldots,t_n$, evaluates to true if the entities denoted by $t_1,\ldots,t_n$ in I are related by the relationship that p denotes in I; else it is false. In other words, we determine whether $\mathrm{I}(t_1),\ldots,\mathrm{I}(t_n) \in \mathrm{I}(p)$.

• If the truth values of wffs α and β are evaluated, then the truth values of the wffs $\neg\alpha$, $\alpha\wedge\beta$, $\alpha\vee\beta$, $\alpha \rightarrow \beta$, and $\alpha \leftrightarrow \beta$ are determined by the usual truth-tables:

α	$\neg\alpha$
T	F
F	T

α	β	$\alpha \wedge \beta$	$\alpha \vee \beta$	$\alpha \rightarrow \beta$	$\alpha \leftrightarrow \beta$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

• $\forall X \alpha$ evaluates to true if the truth value of α is true for *every* object in U; otherwise, it is false.

• $\exists X \alpha$ evaluates to true if the truth value of α is true for *at least one* object in U; otherwise, it is false.

6.2.2 More Formal Presentation

Let's repeat the semantics given above but with a modicum of additional formalism.

Sometimes when being more formal we talk of *models*.[3] These are abstract

[3]Unfortunately, the word 'model', in addition to its everyday meanings, is *often* used in logic in a different way from the way it is being used here. Here, a model is a universe of discourse and an interpretation function, which together allow us to ascribe meaning to wffs

structures used in providing a semantics to a formal language. Specifically, a model, M, consists of a universe of entities, U, and an interpretation function, I:

M = <U, I>

We can now show how to get the semantic value or denotation of a wff. This semantic value or denotation is given with respect to some model, M. Hence the notation used for the semantic value of α with respect to M is $[[\alpha]]^M$:

1. If α is a constant, then $[[\alpha]]^M = \mathrm{I}(\alpha)$.
2. If α is a function symbol, then $[[\alpha]]^M = \mathrm{I}(\alpha)$.
3. If α is a predicate symbol, then $[[\alpha]]^M = \mathrm{I}(\alpha)$.
4. If α is a term comprising an n-place function symbol, f, and terms $t_1, \ldots, t_n$, i.e. α is $f(t_1, \ldots, t_n)$, then $[[\alpha]]^M = [[f]]^M([[t_1]]^M, \ldots, [[t_n]]^M)$ (i.e. you get the function that corresponds to f and the objects corresponding to $t_1, \ldots, t_n$ and apply the function to the objects to get a new object, this being the meaning of the term, $f(t_1, \ldots, t_n)$.)
5. If α is an atom comprising an n-ary predicate, p, and terms $t_1, \ldots, t_n$, i.e. α is $p(t_1, \ldots, t_n)$, then $[[\alpha]]^M = \mathbf{true}$ iff $<[[t_1]]^M, \ldots, [[t_n]]^M> \in [[p]]^M$ (i.e. you get the relation that corresponds to p and the objects corresponding to $t_1, \ldots, t_n$ and see whether this relation does hold for these objects).
6. If α is $\neg\beta$, then $[[\alpha]]^M = \mathbf{true}$ iff $[[\beta]]^M = \mathbf{false}$.
7. If α is $\beta \vee \gamma$, then $[[\alpha]]^M = \mathbf{true}$ iff $[[\beta]]^M = \mathbf{true}$ or $[[\gamma]]^M = \mathbf{true}$.
8. If α is $\beta \wedge \gamma$, then $[[\alpha]]^M = \mathbf{true}$ iff $[[\beta]]^M = \mathbf{true}$ and $[[\gamma]]^M = \mathbf{true}$.
9. If α is $\beta \rightarrow \gamma$, then $[[\alpha]]^M = \mathbf{true}$ iff $[[\beta]]^M = \mathbf{false}$ or $[[\gamma]]^M = \mathbf{true}$.
10. If α is $\beta \leftrightarrow \gamma$, then $[[\alpha]]^M = \mathbf{true}$ iff $[[\beta]]^M = [[\gamma]]^M$.
11. If α is $\forall X\beta$, then $[[\alpha]]^M = \mathbf{true}$ iff $[[\beta]]^M$ is true for all assignments of entities in U to X.
12. If α is $\exists x\beta$, then $[[\alpha]]^M = \mathbf{true}$ iff $[[\beta]]^M = \mathbf{true}$ for at least one assignment of an entity in U to X.

---

and subexpressions of wffs. For this meaning, the word 'structure' or 'conceptualisation' is sometimes used instead. The other meaning (not used here) is that a model is an interpretation that makes a wff true. We do not use the word model in this way in this course. Alan Frisch used the word 'model' with both these meanings in TOC. When he spoke simply of a 'model', he meant a universe and an interpretation function (i.e. the same use as in this course); when he spoke of 'a model of a wff', he meant an interpretation that makes the wff true (i.e. the meaning not used in this course).

Notice that the assumption that we made earlier that our wffs contain no free variables has allowed us a slightly simpler presentation of this material than the one given in TOC. In TOC, free variables were allowed and so one might want to determine, e.g., $[[\mathrm{p}(X)]]^M$. My definition above gives no way of ascribing a meaning to this formula. In TOC, the notion of a **value assignment** that assigns objects from U to the variables in the formula is introduced. Then you determine the meaning of a formula α with respect to a model M and a value assignment g, i.e. $[[\alpha]]^{M,g}$. (In fact, there is another reason for using value assignments: our definitions (11) and (12) above are not recursive definitions, i.e. they do not straightforwardly compute the meaning of an expression from the meaning of its parts. Making them recursive is most often done using value assignments.)

Example

In this example, we use a model, M, in which U = $\{1, 2\}$ and I is defined as follows:

I(a)	=	1
I(f)	=	$\{(1 \mapsto 2), (2 \mapsto 1)\}$, i.e. f corresponds to the function that maps the object 1 to the object 2 and vice versa.
I(p)	=	$\{<2>\}$, i.e. p corresponds to the relation that is true of the object 2, but not the object 1,
I(q)	=	$\{<1, 1>, <1, 2>, <2, 2>\}$

Now we can evaluate some wffs w.r.t. this model.

$[[\mathrm{p(a)} \rightarrow \mathrm{q(a, a)}]]^M$ is true for the following reason:

$[[\mathrm{p(a)}]]^M$ is false because $[[\mathrm{a}]]^M = \mathrm{I(a)} = 1$ is not in the relation $[[\mathrm{p}]]^M = \mathrm{I(p)} = \{<2>\}$. The fact that $[[\mathrm{q(a, a)}]]^M$ is true is irrelevant to the truth of the original wff: when the antecedent of an implication is false, the implication as a whole is true.

$[[\forall X(\mathrm{p}(X) \rightarrow \mathrm{q}(\mathrm{f}(X), \mathrm{a}))]]^M$ is true for the following reason:

If 1 is assigned to X,

then the antecedent is false (1 is not a member of $[[\mathrm{p}]]^M$), so the wff as a whole is true;

If 2 is assigned to X,

then the antecedent is true (2 is a member of $[[\mathrm{p}]]^M$). In the consequent, $[[\mathrm{f}]]^M$ maps 2 to 1, and $[[\mathrm{a}]]^M$ is 1, and $<1,1>$ is a member of $[[\mathrm{q}]]^M$. So the consequent is true. Since the antecedent and the consequent are both true, the implication is true.

Since $[[\mathrm{p}(X) \rightarrow \mathrm{q}(\mathrm{f}(X),\ \mathrm{a})]]^M$ is true for all assignments of objects to X (which we know it is, because we tried them all, and thankfully there were only two of them), then $[[\forall X(\mathrm{p}(X) \rightarrow \mathrm{q}(\mathrm{f}(X),\ \mathrm{a}))]]^M$ is true.

6.2.3 Logical Consequence and Other Metatheoretic Definitions

Once models are defined, we can define satisfiability, unsatisfiability, validity and, most importantly, logical consequence.[4]

Satisfiability and unsatisfiability

Let α be a wff (with no free variables as usual), Φ be a set of wffs, and M be a model.

- If the wff α is true in a model M, i.e. $[[\alpha]]^M$ = true, then M *satisfies* α.
- The wff α is *satisfiable* iff there exists some model M such that α is evaluated to true in M, i.e. $[[\alpha]]^M$ = true. (In this case, M *satisfies* α.) This definition extends to a set of wffs, Φ: the set is satisfiable iff there exists some model, M, which satisfies all the wffs in Φ.
- The wff α is *unsatisfiable* iff there exists no model that satisfies α. (Similarly for a set of wffs.)
- The wff α is *valid* (or *unfalsifiable* or a *tautology*) iff every model satisfies α. We write this

$$\models \alpha$$

Here are some examples:

p(a) is satisfiable

p(a) $\wedge$ ¬p(a) is unsatisfiable (always false)

p(a) $\vee$ ¬p(a) is valid (always true)

Logical Consequence

The wff α is a *logical consequence* of (or an *entailment* of or *logically follows* from or is *logically implied* by) a set of wffs Φ iff every model satisfying Φ also

[4] Beware that TOC used the word 'valid' in two ways, of which we give only one.

satisfies α. (Or, if you like, there is no model that makes all the elements of Φ true but makes α false.) We write this

$$\Phi \models \alpha$$

Here's an example.

$$\{\forall X(\mathrm{p}(X) \rightarrow \mathrm{q}(X)), \mathrm{p}(\mathrm{a})\} \models \mathrm{q}(\mathrm{a})$$

It does not matter what p, q and a are interpreted as, *if* the members of the set $\{\forall X(\mathrm{p}(X) \rightarrow \mathrm{q}(X)), \mathrm{p}(\mathrm{a})\}$ *are* true in M, then q(a) *will* be true in M. Hence, $\{\forall X(\mathrm{p}(X) \rightarrow \mathrm{q}(X)), \mathrm{p}(\mathrm{a})\} \models \mathrm{q}(\mathrm{a})$.

An informal demonstration of this follows. Suppose in one model p denotes the set of men, q denotes the set of mortals and a denotes Socrates. Then if it is the case that all men are mortals and Socrates is a man (i.e. if $\{\forall X(\mathrm{p}(X) \rightarrow \mathrm{q}(X)), \mathrm{p}(\mathrm{a})\}$ is true) then it follows that Socrates is mortal (i.e. q(a) is true). Suppose in another model p denotes the set of elephants, q denotes the set of grey things and a denotes Clyde. Then if it is the case that all elephants are grey and Clyde is an elephant (i.e. if $\{\forall X(\mathrm{p}(X) \rightarrow \mathrm{q}(X)), \mathrm{p}(\mathrm{a})\}$ is true) then it follows that Clyde is grey (i.e. q(a) is true). Suppose in yet another model p denotes the set of elephants, q denotes the set of yellow things and a denotes Clyde. Then if it is the case that all elephants are yellow (never mind that it isn't in our world: *if* it *is* the case...) and Clyde is an elephant, then it follows that Clyde is yellow. And so on.

Note how the definition of logical consequence refers to *all* models which satisfy Φ. This suggests that in order to determine entailments of a set of wffs we would have to check all interpretations of the wffs for all universes. This is possible in some logics. For example, we can do it for propositional logic (i.e. logic without variables and functions). In propositional logic, checking all interpretations is the same as constructing a truth-table since each row of the table is an interpretation. For example, we can prove that q logically follows from the set $\{\neg\mathrm{p}, \mathrm{p} \vee \mathrm{q}\}$:

p	q	¬p	p ∨ q
T	T	F	T
T	F	F	T
F	T	T	T
F	F	T	F

q logically follows from $\{\neg\mathrm{p}, \mathrm{p} \vee \mathrm{q}\}$ because it is true in all cases (interpretations) in which ¬p and p ∨ q are true (i.e. row 3).

In other logics, determining logical consequences cannot in general be done by inspecting models. For example it is not possible in FOPC because there may be an infinite number of interpretations (for an infinite number of possibly infinite universes). We clearly need an alternative approach.

6.3 The Proof Theory of First Order Predicate Calculus

6.3.1 Informal Presentation

The *proof theory* (or proof system) of a logic is its formal deduction system. Given a set of wffs, Φ (the premisses), the proof theory enables us to *derive* or *deduce* conclusions from Φ by *syntactic operations* alone. In other words, the proof theory tells us how to structurally manipulate Φ, without reference to truth values, in such a way as to derive new formulas which (if the manipulation rules are 'right') are guaranteed to be logical consequences (entailments) of Φ.

By proceeding entirely by symbol manipulation, a deduction makes no use at all of semantic concepts such as truth in a model, any more than the rules of chess require an understanding of medieval cavalry battles. This is pretty handy as far as computers go.

The proof theory of a logic comprises a finite set of *inference rules* and a finite set of *logical axiom schemata.*

An *inference rule* comprises a set of patterns called *conditions* and another set of patterns called *conclusions.* As an example, if α and β are wffs, then in the inference rule Modus Ponens, the conditions are α and $\alpha \to \beta$; the conclusion is β. If we have wffs that match the conditions, then we can infer wffs that match the conclusions. For example, if we have the wffs

raining $\to$ (need-mac $\vee$ need-brolly)

and

raining,

which do match the conditions of Modus Ponens, then we can infer the wff that matches the conclusion, namely

need-mac $\vee$ need-brolly.

Some examples of logical axiom schemata (where α, β and γ are wffs) are:

$$\alpha \to (\beta \to \alpha)$$
$$(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$$
$$(\alpha \to \neg\beta) \to ((\alpha \to \beta) \to \neg\beta)$$
$$(\neg\alpha \to \neg\beta) \to ((\neg\alpha \to \beta) \to \alpha)$$

If you consistently substitute wffs for α, β and γ in these logical axiom schemata, you get *logical axioms*, i.e. wffs that are valid (satisfied in all models). Logical axiom schemata are thus 'templates' for tautologies. For example, if you substitute raining

for α and

(need-mac $\vee$ need-brolly)

for β in the first logical axiom schemata, you produce the logical axiom

raining $\rightarrow$ ((need-mac $\vee$ need-brolly) $\rightarrow$ raining).

You can confirm that this is a tautology by drawing up a truth-table and noting that the wff always evaluates to true. (Exercise!)

To use a proof theory to prove things, you supply *a set of proper axioms*, which are wffs which (under appropriate interpretations) make statements about the situation in which you are interested.

Given a set of inference rules, a set of logical axiom schemata and a set of proper axioms, Φ, a *proof* or *derivation* of some wff α from Φ is defined to be a finite sequence of wffs $\beta_1, \beta_2, \ldots, \beta_n$ such that $\alpha = \beta_n$ and for each i $(1 \leq i \leq n)$ either

(a) β_i is a logical axiom (produced by substituting wffs into the logical axiom schemata), or

(b) β_i is a proper axiom (a member of Φ), or

(c) β_i is the result of applying an inference rule to previous wffs in the sequence.

If there exists a proof of α from Φ then we say that α is *derivable* (or *provable* or *deducible*) from Φ. Symbolically we write this as

$$\Phi \vdash \alpha$$

6.3.2 Relating Proof Theory to Model Theory

To some extent you can pick whatever set of logical axiom schemata and inference rules you like when defining a proof theory. But this is not a good idea. As an analogy, we might have some pebbles and some rules, where one of the rules tells us to push two heaps of pebbles together to form a bigger heap. We cannot call this "multiplication" since often it will get the wrong result: to get the right result we need the right rules (in this case a more specific rule). We must choose rules which manipulate the symbols 'correctly' according to the meaning we give them. There are two measures of this: *soundness* and *completeness*.

Soundness: A proof theory is *sound* if $\Phi \vdash \alpha$ implies $\Phi \models \alpha$, i.e. if all the wffs that can be derived from Φ are also logical consequences/entailments of Φ, i.e. the proof theory produces no rubbish.

Completeness: A proof theory is *complete* if $\Phi \models \alpha$ implies $\Phi \vdash \alpha$, i.e. every entailment from Φ can be derived from Φ, i.e. the proof theory produces everything the model theory does.

6.4 A Proof Theory for FOPC

Here we present a so-called *Hilbert system.* Note that this is only one of many possible proof systems for FOPC (e.g. Ben-Ari also presents a Gentzen system and future worksheets present refutation resolution on clausal form logic). I shall present only the parts for propositional logic, rather than giving the full complexity needed for FOPC.

This system has the following components:

Logical axiom schemata: for wffs α, β and γ,
Axiom schemata 1: $(\neg\alpha \to \alpha) \to \alpha$,
Axiom schemata 2: $\alpha \to (\neg\alpha \to \beta)$,
Axiom schemata 3: $(\alpha \to \beta) \to ((\beta \to \gamma) \to (\alpha \to \gamma))$

Inference rules: there is only one rule, Modus Ponens, which states for wffs α and β,:

$$\frac{\vdash \alpha \qquad \vdash \alpha \to \beta}{\vdash \beta}$$

For your proper axioms you must restrict yourself to using $\to$ and $\neg$ only. (This is no loss since a wff that contains the other connectives can always be converted into an equivalent wff using only the given three.)

Here is an example of a proof using this theory. Given that if it is raining then I get wet, and if I get wet then I catch cold, show that if it is raining then I catch cold:

1.	rain $\to$ wet	(proper axiom)
2.	wet $\to$ cold	(proper axiom)
3.	(rain $\to$ wet) $\to$ ((wet $\to$ cold) $\to$ (rain $\to$ cold))	(Substitution of rain for α, wet for β and cold for γ in logical axiom schemata 3)
4.	((wet $\to$ cold) $\to$ (rain $\to$ cold))	(by Modus Ponens using 1 and 3)
5.	(rain $\to$ cold)	(by Modus Ponens using 2 and 4)

You can see the role of the logical axiom schemata: they can help us to construct fodder for the inference rules.

The proofs are quite long even in what look like simple cases. In fact, a Hilbert system is usually made more useful by adding extra inference rules, where these extra rules are derived from the original rules and logical axiom schemata but allow what would have taken several steps to be achieved with a single step, i.e. they add no extra power, they just shorten the proofs. One example is the *deduction rule*, which says that to prove $\alpha \to \beta$, then if you *assume* α has already been proven, and can derive β (probably making use of the assumption of α in doing so), then you can conclude $\alpha \to \beta$. More formally, this is written as:

$$\frac{\Phi \cup \{\alpha\} \vdash \beta}{\Phi \vdash \alpha \to \beta}$$

where α and β are wffs and Φ is a set of wffs.

The difficulty with these proofs is that you are faced with so many possible things that you can try. You have to *search* for a derivation by trial and error. Similarly, the only way a computer is going to find a derivation is by *search*. But there is a problem: the search space is huge. Just think about this: if I have two proper axioms, p and q, then just from the logical axiom schemata

$$(\alpha \to \beta) \to ((\beta \to \gamma) \to (\alpha \to \gamma))$$

I get 8 new wffs by using every combination of the proper axioms p and q for α, β and γ. Each of these 8 can then be plugged in, again in any combination, to give a further 992 possibilities, and so on. That's an enormous search space for a trivial example. Clearly more complex examples with more proper axioms are even worse. And when the other logical axiom schemata and inference rules are considered the situation becomes horrendous.

Blind search will be an inadequate implementation. We would have to come up with some very clever search algorithms (ones that use sophisticated heuristics). So far researchers haven't come up with these. What they've done instead is move to other proof theories. In particular, we will now look at one which has only one inference rule and no logical axiom schemata.

6.5 Exercises

Exercise 6.1 *Let $c(X)$ mean 'X is a used-car dealer' and $h(X)$ mean 'X is honest'. Translate the following into English:*

1. $\exists X \; c(X)$
2. $\exists X \; h(X)$
3. $\forall X \; (c(X) \to h(X))$
4. $\forall X \; (c(X) \wedge h(X))$
5. $\exists X \; (c(X) \wedge h(X))$
6. $\exists X \; (c(X) \to h(X))$

7. $\forall X\ (c(X) \rightarrow \neg\ h(X))$

Exercise 6.2 *Let $b(X)$ mean 'X is a book', $a(X,Y)$ mean 'X is the author of Y', $e(X)$ mean X is expensive, wg mean William Golding and ts mean 'The Spire'. Using these predicate and constant symbols and no others, translate the following sentences of English into FOPC:*

1. *Some books are expensive.*
2. *All books are expensive.*
3. *Not all books are expensive.*
4. *All books have an author.*
5. *If 'The Spire' is a book, then William Golding wrote it.*
6. *William Golding wrote 'The Spire' only if it is a book.*

Exercise 6.3 *Let ratl(X) and real(X) mean 'X is a rational number' and 'X is a real number' respectively. Translate the following into FOPC:*

1. *Every rational number is a real number.*
2. *Some real numbers are rational numbers.*
3. *Not every real number is a rational number.*

Exercise 6.4 *Let eq(X,Y) mean 'X is equal to Y', let 0 be the number 0, and let succ(X) and pred(X) be function symbols that represent the immediate successor and predecessor of X. Translate the following into English (make your translations as natural as possible):*

1. $\forall X \exists Y\ (eq(succ(X), Y) \wedge \forall Z\ (eq(succ(X), Z) \rightarrow eq(Y, Z)))$
2. $\neg\exists X\ eq(0, succ(X))$
3. $\forall X\ (\neg eq(X, 0) \rightarrow (\exists Y\ (eq(pred(X), Y) \wedge \forall Z\ (eq(pred(X), Z) \rightarrow eq(Y, Z))))$

Exercise 6.5 *For the model, M, given in the body of this worksheet (i.e the one in which $U = \{1, 2\}$, etc.), evaluate the following wffs.*

(1) $[[p(a) \wedge q(a, a)]]^M$
(2) $[[\neg p(f(a)) \vee q(a, f(a))]]^M$
(3) $[[\exists X\, (p(f(X)) \wedge q(X, f(a)))]]^M$
(4) $[[\exists X\, (p(X) \wedge q(X, a))]]^M$
(5) $[[\forall X \exists Y\, q(X, Y)]]^M$
(6) $[[\exists Y \forall X\, q(X, Y)]]^M$

Acknowledgements

Exercises have been based on (or in some cases directly lifted from) Chang & Lee.