7 Classical Quantified Logic

Before examining quantified modal logic, we review a bare-bones system for first-order predicate or quantificational logic, i.e., that involving the quantifiers $\exists x \ldots$ and $\forall x \ldots$, and so on.

7.1 Syntax

It is worthwhile to set out the syntax of the language in full, recursively, giving precise definitions. This is especially important if we plan on doing metatheory for systems for quantified logic.

A variable is any lowercase letter from x to z, written with zero or more occurrences of $'$, i.e.: $\mathrm{x}, \mathrm{x}', \mathrm{x}'', \mathrm{x}''', \ldots, \mathrm{y}, \mathrm{y}', \mathrm{y}'', \mathrm{y}''', \ldots, \mathrm{z}, \mathrm{z}', \mathrm{z}'', \mathrm{z}''', \ldots$

A constant is any lowercase letter from a to r, written with zero or more occurrences of $'$, i.e.: $\mathrm{a}, \mathrm{a}', \mathrm{a}'', \mathrm{a}''', \ldots, \mathrm{b}, \mathrm{b}', \mathrm{b}'', \mathrm{b}''', \ldots, \mathrm{c}, \mathrm{c}', \mathrm{c}'', \mathrm{c}''', \ldots$

A term is a constant, or any other expression to which we may instantiate variables (at the moment, all our terms are constants, but we may later add descriptions, function terms, etc.)

A predicate letter is any uppercase letter from E to T, written with zero or more occurrences of $'$, i.e.: $\mathrm{F}, \mathrm{F}', \mathrm{F}'', \mathrm{F}''', \ldots, \mathrm{G}, \mathrm{G}', \mathrm{G}'', \mathrm{G}''', \ldots$

In stating general inference rules, schemata and other general results, it is useful to have schematic variables in the metalanguage to represent any arbitrary sign in one of these categories.

For this purpose, we use signs similar to the above, but italicized, so that "x" for example, means *any* variable, not necessarily "x", "P" means *any* predicate, not necessarily "P", and so on. These may look so similar you might not be able to remember the difference, but in most ways, this is harmless if not helpful. (Garson completely ignores the need for a distinction here.)

We shall also use the letters t, s, u and v in the metalanguage schematically for *any term* of the object language.

We continue to use A, B, C and D schematically for any complicated expression of the object language, typically restricting their use to sentences (defined below) or open sentences.

In the metalanguage, we write "$[A]^t/x$" to mean the expression that results when term t is substituted for all free occurrences of the variable x in A. For example, if A is "F(y) & R(y, b)", t is "a", and x is "y", then $[A]^t/x$ is "F(a) & R(a, b)".

That notation can be cumbersome. We shall also sometimes use "Ax" or "$A(x)$" for an arbitrary formula containing x free, and then, if "At" is used in the same context, it means the result of substituting t for all free occurrences of x. Note, however, that Ax may contain x in more than one spot, and may contain variables other than x. Therefore, we shall sometimes require the more sophisticated, but awkward sort of notation above.

An atomic sentence is either $\perp$ or something of the form $P(t_1, t_2, \ldots, t_n)$, where P is a predicate letter and $t_1, t_2, \ldots, t_n$ are any (possibly zero) number of terms.

In the case where $n = 0$, an atomic sentence would look something like "Q()"; these may be treated just like atomic statement letters for propositional logic.

We allow ourselves to leave off the parentheses and commas in an atomic sentence if doing so is unlikely to create any confusion. Thus instead of "R(a, b)" we may write "Rab". Instead of "Q()" we may write "Q", and so on.

A sentence of the language is defined recursively:

1. Any atomic sentence is a sentence.
2. If A and B are sentences, then $(A \to B)$ is a sentence.
3. If Ac is a sentence containing some constant c, then $\forall x\, Ax$ is a sentence for any variable x.
4. *(For quantified modal logic, we add:)* If A is a sentence, then $\Box A$ is a sentence.

If the main operator of a sentence is "$\to$", we shall allow ourselves to omit the matching outer parentheses as a convenience. $\sim$, &, $\vee$, $\leftrightarrow$, $\Diamond$ and $\dashv$ are defined as before.

The existential quantifier is introduced not as a primitive operator, but as an abbrevation:

(Def $\exists$) $\quad \exists x\, Ax \quad$ *abbreviates* $\quad \sim\forall x \sim Ax$

You may be used to different notation:

Quantifier	Garson	Alternatives
Universal	$\forall x$	$(x), (\forall x), \wedge_x, \Pi_x$
Existential	$\exists x$	$(\exists x), \vee_x, \Sigma_x$

7.2 Classical Inference Rules

All the primitive inference rules from system PL ((MP), (CP), (Reit), (DN), etc.) are carried over to all of our quantified systems. All the derived rules that follow from these rules (e.g., (MT), (DN), ($\bot$In), ($\vee$Out) etc.) carry over as well, and may be used in your proofs.

The classical inference rules for quantification theory can be stated as follows:

(Q$\forall$Out) From $\forall x\, Ax$ infer Ac.

(Q$\forall$In) Where no premises or undischarged hypotheses of the subproof contain the constant c, from Ac infer $\forall x\, Ax$, provided that c does not occur in $\forall x\, Ax$ and c does not already occur in Ac within the scope of a quantifier binding the variable x.[1]

(Q$\forall$Out) is standard universal instantiation.

(Q$\forall$In) codifies the idea that if a result can be proven about an arbitrary individual, one you have no special knowledge about or are making any special assumptions about (named by a constant not occurring in your hypotheses), then it must hold true for every individual. This is akin to Hardegree's UD or Copi/Hurley's UG.

System QL- is the system containing the primitive inference rules of PL (except in a language with our new syntax) along with the above two rules.

An example derivation. Here we assume that B is a sentence not containing x free.

($\exists$-Lemma) $\exists x\, Ax, \forall x(Ax \to B) \vdash_{\text{QL-}} B$

1.	$\exists x\, Ax$	
2.	$\forall x(Ax \to B)$	
3.	$\sim\forall x \sim Ax$	1 (Def $\exists$)
4.	$\sim B$	
5.	$Ac \to B$	2 (Q$\forall$Out)
6.	$\sim Ac$	4,5 (MT)
7.	$\forall x \sim Ax$	6 (Q$\forall$In)
8.	$\bot$	3,7 ($\bot$In)
9.	B	4–8 (IP)

We were able to infer the universal statement at line 7 because c does not occur in our assumptions at line 1, 2 and 4. (If some constants do occur there, we choose c as the first constant alphabetically that does not.)

The above result is useful for obtaining, as derived rules, the usual rules for $\exists$:

(Q$\exists$In) From Ac infer $\exists x\, Ax$. *Proof scheme:*

Ac	
$\forall x \sim Ax$	
$\sim Ac$	(Q$\forall$Out)
Ac	(Reit)
$\bot$	($\bot$In)
$\sim\forall x \sim Ax$	(IP)
$\exists x\, Ax$	(Def $\exists$)

(Q$\exists$Out) If one has a statement of the form $\exists x\, Ax$, and from Ac one can derive B, where c is a constant not occurring in B or $\exists x\, Ax$ or any hypotheses on which $\exists x\, Ax$ depends, then one may infer B. *Proof scheme:*

$\exists x\, Ax$	
Ac	
$\vdots$	
B	
$Ac \to B$	(CP)
$\forall x(Ax \to B)$	(Q$\forall$In)
B	($\exists$-Lemma)

Henceforth this gets abbreviated as follows:

$\exists x\, Ax$	
Ac	for a new constant c
$\vdots$	
B	
B	(Q$\exists$Out)

If you prefer you may forgo the subproof and annotate instead as:

$\exists x\, Ax$		
Ac	(Q$\exists$Out)	for a new constant c
$\vdots$		
B		

[1] Garson leaves out this last stipulation. This is presumably a mistake since it is necessary for the soundness of the system, e.g., to avoid inferring $\forall x(Fx \to \exists x\, Rxx)$ from $Fa \to \exists x\, Rxa$.

However, you must be careful not to end the proof with a line that contains c.

Here are some more useful derived rules:

(∼∃Out) From $\sim\exists x\, Ax$ infer $\forall x \sim Ax$. (Really just DN.)

(∼∃In) From $\forall x \sim Ax$ infer $\sim\exists x\, Ax$. (Really just DN.)

(∼∀Out) From $\sim\forall x\, Ax$ infer $\exists x \sim Ax$. *Proof:*

$\sim\forall x\, Ax$	
| $\forall x \sim\sim Ax$	
| $\sim\sim Ac$	(Q∀Out)
| Ac	(DN)
| $\forall x\, Ax$	(Q∀In)
| $\perp$	(⊥In)
$\sim\forall x \sim\sim Ax$	(IP)
$\exists x \sim Ax$	(Def ∃)

(∼∀In) From $\exists x \sim Ax$ infer $\sim\forall x\, Ax$. *Proof:*

$\exists x \sim Ax$	
$\sim\forall x \sim\sim Ax$	(Def ∃)
| $\forall x\, Ax$	
| Ac	(Q∀Out)
| $\sim\sim Ac$	(DN)
| $\forall x \sim\sim Ax$	(Q∀In))
| $\perp$	(⊥In)
$\sim\forall x\, Ax$	(IP)

Most likely, this gives you all the inference rules you learned for the quantifiers.

7.3 Identity logic

The "-" in the name of system QL- indicates the exclusion of identity logic. To get the full system QL, we incorporate identity logic.

(Def =)	$t = s$	*abbreviates*	I(t, s)
(Def ≠)	$t \neq s$	*abbreviates*	∼I(t, s)

Note that here "I" is just a particular predicate letter that we have set aside for identity, but introduce the standard, more recognizable notation as a convenience.

Garson uses the notation $t \approx s$ instead for identity. I don't know why, and I found the deviation from standard practice too annoying to follow.

QL adds to QL- the following axiom schema (=In) and inference rule (=Out):

(=In) $t = t$

(=Out) From $s = t$ and $P(u_1, u_2, \ldots, u_n, s, v_1, v_2, \ldots, v_m)$ infer $P(u_1, u_2, \ldots, u_n, t, v_1, v_2, \ldots, v_m)$, where P is any predicate letter, and $u_1, u_2, \ldots, u_n$ and $v_1, v_2, \ldots, v_m$ are (possibly empty) lists of terms going together with s to form the atomic sentence $P(u_1, u_2, \ldots, u_n, s, v_1, v_2, \ldots, v_m)$.

(=In) is the standard "rule" (though really an axiom schema) of the reflexivity of identity. (=Out) is similar to the rule often called "Leibniz's law" (LL) or "the substition of identicals", except that it only allows for substitutions within atomic sentences. It also only allows for one substitution at a time, though with repeated uses, one can make multiple substitutions.

From it, we get the following derived rules:

(Sym=) From $s = t$ infer $t = s$. *Proof*:

$s = t$	
$s = s$	(=In)
I(s, s)	(Def =)
I(t, s)	(=Out)
$t = s$	(Def =)

(Trans=) From $s = t$ and $t = u$ infer $s = u$. *Proof*:

$s = t$	
$t = u$	
I(t, u)	(Def =)
$t = s$	(Sym=)
I(s, u)	(=Out)
$s = u$	(Def =)

While (=Out) does not have the full strength of (LL), it is possible, from (=Out), to infer At from As and $s = t$ for any extensional statement of the language of QL, one without modal or other intensional operators. (This can be proven by wff induction on the length of of As.) Just as an example we can substitute identicals within negations of atomic statements by means of an indirect proof:

$s = t$	
$\sim P(s)$	
$\quad P(t)$	
$\quad t = s$	(Sym=)
$\quad P(s)$	(=Out)
$\quad \bot$	($\bot$In)
$\sim P(t)$	(IP)

If you'd like, you may cite (LL) for substituting identicals in any context not involving modal operators.

7.4 Modal Extensions of QL

We do not need to do anything special to develop systems for modal logic based on classical quantificational theory; one needs only add the rules of a given propositional modal logic (modified only slightly to accommodate our new syntax). Garson prefixes the name of a modal logic system with "q" to indicate its version within classical quantification theory. Thus:

System qK: The system obtained from QL by adding "□" to the syntax and (□In) and (□Out) as additional inference rules.

System qT: The system obtained from qK by adding every instance of schema (M) as an axiom.

System qB: The system obtained from qT by adding every instance of schema (B) as an axiom.

System qS4: The system obtained from qT by adding every instance of schema (4) as an axiom.

System qS5: The system obtained from qT by adding every instance of schema (5) as an axiom.

Of course, we do need to reinterpret what we mean by "every instance of (M)," etc., to accommodate our expanded syntax. Thus, we have instances such as:

$\Box\forall xFx \rightarrow \forall xFx$
$\Box a = b \rightarrow a = b$
$\Box\Diamond\exists x\Box\forall yRxy \rightarrow \Diamond\exists x\Box\forall yRxy$

All the derived rules we proved for the propositional versions of K, T, B, S4 and S5 carry over to their quantified versions, including, e.g., (◇Out).

As an example, let us show the following:

$\vdash_{qK} \forall x(\Box Fx \leftrightarrow \Box\forall y(x = y \rightarrow Fy))$

1.	$\quad \Box Fc$	
2.	$\quad\quad \Box$	
3.	$\quad\quad Fc$	(□Out)
4.	$\quad\quad\quad c = d$	
5.	$\quad\quad\quad Fd$	3,4 (=Out)
6.	$\quad\quad c = d \rightarrow Fd$	4–5 (CP)
7.	$\quad\quad \forall y(c = y \rightarrow Fy)$	6 (Q∀In)
8.	$\quad \Box\forall y(c = y \rightarrow Fy)$	2–7 (□In)
9.	$\Box Fc \rightarrow \Box\forall y(c = y \rightarrow Fy)$	1–8 (CP)
10.	$\quad \Box\forall y(c = y \rightarrow Fy)$	
11.	$\quad\quad \Box$	
12.	$\quad\quad \forall y(c = y \rightarrow Fy)$	10 (□Out)
13.	$\quad\quad c = c \rightarrow Fc$	12 (Q∀Out)
14.	$\quad\quad c = c$	(=In)
15.	$\quad\quad Fc$	13,14 (MP)
16.	$\quad \Box Fc$	11–15 (□In)
17.	$\Box\forall y(c = y \rightarrow Fy) \rightarrow \Box Fc$	10–16 (CP)
18.	$\Box Fc \leftrightarrow \Box\forall y(c = y \rightarrow Fy)$	9,17 (↔In)
19.	$\forall x(\Box Fx \leftrightarrow \Box\forall y(x = y \rightarrow Fy))$	18 (Q∀In)

As with the other rules, the rules for quantifiers and identity must be applied within the same subproof. Hence, we can see how Leibniz's law does not hold in all contexts. Using "G" for "greater than", "e" for "eight", "f" for "five", and "n" for "the number of planets", we can see what blocks the infernce from "□G(e, f)" and "n = e" to "□G(n, f)".

1.	$\quad \Box G(e, f)$	
2.	$\quad n = e$	
3.	$\quad\quad \Box$	
4.	$\quad\quad G(e, f)$	1 (□Out)

Here one cannot infer G(n, f) since we do not have n = e inside the boxed subproof. (Notice, however, that the proof would go through if we had □n = e instead at line 2.

7.5 Dissatisfaction with Classical Quantified Logic

Despite its naturalness, historical importance, and near dominance outside of philosophy (e.g., in mathematics), classical quantified logic has its detractors, especially for modal logic.

Classical quantifier rules assume that every term denotes something in the domain of the quantifiers, so that from Ac one can always infer $\exists x\, Ax$.

At the very least, this complicates translating statements or chains of reasoning from natural language in which names that don't refer (or might not refer). As Garson points out, the following "proof" of the existence of God does not seem compelling:

1.	$\mathrm{g} = \mathrm{g}$	(=In)
2.	$\exists \mathrm{x}\, \mathrm{x} = \mathrm{g}$	(Q∃In)

Similarly, it seems that from the truth that no horses have wings, classical quantified logic allows us to infer (falsely, one might that) that Pegasus is not a winged horse.

1.	$\forall \mathrm{x}(\mathrm{Hx} \rightarrow \sim\mathrm{Wx})$	
2.	Hp & Wp	
3.	Hp	2 (&Out)
4.	Wp	2 (&Out)
5.	$\mathrm{Hp} \rightarrow \sim\mathrm{Wp}$	1 (Q∀Out)
6.	$\sim\mathrm{Wp}$	3,5 (MP)
7.	$\perp$	4,6 (⊥In)
8.	$\sim$(Hp & Wp)	2–7 (IP)

The explanation one might give is that a quantified statement such as $\forall \mathrm{x}(\mathrm{Hx} \rightarrow \sim\mathrm{Wx})$ is only about existent things, not unreal things like Pegasus.

Of course, these are not the sorts of arguments that a defender of classical quantified logic would take lying down.

FREGE, the progenitor of modern quantificational logic, regarded it as a *defect* of ordinary language that it contained names that did not refer. Frege thought that any sentence containing a nondenoting expression lacked a truth-value altogether, and hence, that logic didn't properly apply to them.

In his own logical language, or "Begriffsschrift," Frege required that a name or other term only be used if it could be shown to denote something. The harshness of this requirement was ameliorated in two ways. First, Frege thought that within the scope of intensional operators like "Homer believed that Apollo is wise," names refer to their customary *senses* or meanings. Hence he would not regard this as neither true nor false simply because "Apollo" does not denote; instead, he would regard that name, in that context, as denoting an abstract sense. This still leads to complications, however, as ordinary names like "Socrates" occuring in such contexts would need to be translated using different signs (signs for senses) from the signs used to translate the same names occurring outside of such contexts.

Second, Frege introduced a means for capturing descriptions, which he wrote "$\backslash\grave{\alpha}(A(\alpha))$", read "the α such that $A(\alpha)$". In case there was one and only one such thing, then this denoted what we would expect. If there were no things such that $A(\alpha)$, Frege regarded "$\backslash\grave{\alpha}(A(\alpha))$" as denoting the empty class, and if there were many, Frege regarded "$\backslash\grave{\alpha}(A(\alpha))$" as denoting the class of all of them. This allowed Frege to ensure that all terms in his logical language could, technically speaking, be treated as denoting.

Criticizing Frege's approach as "artificial," RUSSELL made a distinction between what he called "logically proper names," and ordinary proper names. A logically proper name stood for an entity one was directly acquainted with, and hence, was guaranteed to refer. Only these would have been translated by him using such constants as "a", "b", or "c". An ordinary proper name like "Socrates" or "Apollo" he regarded as a disguised definite description (e.g., "the teacher of Plato" or "the sun god", etc.)

Moreover, unlike Frege, Russell did not regard descriptins as genuine terms. According to Russell, a description, "the x such that Ax", which he wrote as "$(\imath x)Ax$", was an "incomplete symbol"; something that contributed to the meaning of a sentence in which it occurs without having a meaning of its own. In particular:

$$B((\imath x)Ax) \quad \textit{abbreviates} \quad \exists x(\forall y(Ay \leftrightarrow y = x)\ \&\ Bx)$$

Russell would not have translated names like "Apollo", "Pegasus" or "God" as simple constants, but as descriptions to be eliminated in context as the above. Hence, "Apollo is wise" might become (where "Sx" means "x is god of the sun", etc.)"

$$\exists \mathrm{x}(\forall \mathrm{y}(\mathrm{Sy} \leftrightarrow \mathrm{y} = \mathrm{x})\ \&\ \mathrm{Wx})$$

This says roughtly that there is one and only one god of the sun, which is wise.

Another advantage of Russell's theory is not seems to eliminate the need to restrict applying (LL) in modal contexts. Since "the number of planets" is a description, it would not count as a term for Russell, and it would be inappropriate to represent it as "n". Hence, "the number of planets = 8" is not a statement of the form "n = e", but rather of the form:

$\exists x(\forall y(Ny \leftrightarrow y = x) \;\&\; x = e)$

From this and "$\Box G(e, f)$" and (LL) one can only obtain:

$\exists x(\forall y(Ny \leftrightarrow y = x) \;\&\; \Box G(x, f))$

But not the more problematic:

$\Box \exists x(\forall y(Ny \leftrightarrow y = x) \;\&\; G(x, f))$

The former says that there is something that alone numbers the planets and it is necessarily greater than five. The latter says that there is necessarily something that alone numbers the planets and is greater than five. (It should be noted that Russell himself did not apply his theory to modal contexts, though he did to belief contexts. Russell had idiosyncratic views on modality.)

Despite its advantages, many remain unconvinced. One difficulty is that Russell's theory makes statements of the form "the F is G" false whenever there isn't a unique F. Taking "$(\iota x)Px$" as the description corresponding to the name "Pegasus", on Russell's analysis, "Pegasus is a horse" becomes:

$\exists x(\forall y(Py \leftrightarrow y = x) \;\&\; Hx)$

This is false. Since many would regard "Pegasus is a horse" as true, Russell's analysis is sometimes seen as flawed.

An alternative approach is *Free Logic*, which allows nondenoting terms. The disadvantage to Free Logic is that it requires altering the rules of inference of classical quantified logic. (The precise details depend on the formulation; we shall see Garson's formulation shortly.)

Whatever one might think of the arguments against classical quantified logic as considered above, there are additional concerns prompting us in the direction of Free Logic for modal logic. For example, it is one thing to conclude, like Frege, that "Pegasus is a horse" is neither true nor false or outside the scope of logic, or to conclude, like Russell, that it is false, but it is quite another to say the same for "It is possible that Pegasus is a horse."

When modal logics are formulated using classical quantifier rules, one in effect gets the result that the things that do exist are the only things that may exist, and that those things that do exist necessarily exist. This is borne out in that the so-called *Barcan Formula* and *Converse Barcan Formula* are theorems of qK. (These are named after logician Ruth Barcan Marcus, who first put them forth for consideration.) The former states that if everything is necessarily some way, then necessarily, everything is that way. The latter states that if necessarily, everything is some way, then everything is necessarily that way.

(BF) $\vdash_{qK} \forall x\, \Box Ax \to \Box \forall x\, Ax$

1.	$\forall x\, \Box Ax$	
2.	$\Box Ac$	1 (Q∀Out)
3.	$\Box$	
4.	Ac	2 (□Out)
5.	$\forall x\, Ax$	4 (Q∀In)
6.	$\Box \forall x\, Ax$	3–5 (□In)
7.	$\forall x\, \Box Ax \to \Box \forall x\, Ax$	1–6 (CP)

(CBF) $\vdash_{qK} \Box \forall x\, Ax \to \forall x\, \Box Ax$

1.	$\Box \forall x\, Ax$	
2.	$\Box$	
3.	$\forall x\, Ax$	1 (□Out)
4.	Ac	3 (Q∀Out)
5.	$\Box Ac$	2–4 (□In)
6.	$\forall x\, \Box Ax$	(Q∀In)
7.	$\Box \forall x\, Ax \to \forall x\, \Box Ax$	1–6 (CP)

To see that these are not as unproblematic as they seem, consider the following. For (BF), suppose that our world is a physicalist one, i.e., that everything is physical. Suppose also that everything that is physical is necessarily physical. Can we conlude from the fact that everything is necessarily physical that necessarily, everything is physical (i.e., that immaterial minds, for example, are impossible?) It seems not.

For (CBF), simply let "Ax" mean "x exists". Necessarily, whatever there is exists. However, does this mean that the things that exist necessarily exist? One might think not. Or, consider other meanings of $\Box$. Suppose I believe that everything is good; does this mean that, of everything, I believe it is good? What of things that I don't know of or have never thought about?

This considerations just advanced are hardly decisive, but at least they might give us impetus to explore the *possibility* of systems of modal logic where the above do not hold.