Testing Validity: QL and Natural Deduction

In essence, natural deduction and semantic trees work in QL as they do in PL. But we need new rules for the quantifiers. Some of these rules are a little difficult—they have to be in order to guarantee derivations that *preserve truth*. On the right are the rules for natural deduction (the syntactic method).

	∀x	∃x
Add Quantifier	Universal Generalisation ∀Intro or ∀I	Existential Generalisation ∃Intro or ∃I
Remove Quantifier	Universal Instantiation ∀Elim or ∀E	Existential Instantiation ∃Elim or ∃E

∀Elim

If $\forall xFx$, then Fa & Fb & Fc & Fd, etc., i.e. if everything in our universe of discourse (UD) is F, then we can pick *any* object and say of it that it is F. Here are two ways to graphically represent the rule.

Metalanguage	*QL*	
⋮	⋮	
$\forall v\phi$	$\forall x(Fx \,\&\, Gx)$	$[v = x]$
$\phi\,[c/v]$	$Fa \,\&\, Ga$	$[c = a]$

Note. Remove the quantifier and then replace all *free(d)* occurrences of v *systematically* and *uniformly* with the constant.

∃Intro

If Fa in our UD, then we can deduce that there is at least something in UD that is F. This is in fact a weaker statement.

Metalanguage	*QL*	
⋮	⋮	
ϕc	$Fa \supset Ga$	$[c = a]$
$\exists v\phi\,[v/c]$	$\exists x(Fx \supset Gx)$	$[x = v]$

Note. In a proof where the conclusion is an existentially quantified sentence, this is (often) the last rule applied.

∀Intro

This is more difficult. We cannot just say, $Fa \vdash \forall xFx$ (e.g., deduce from 'Plato smiles' that 'everyboy smiles'). Induction is out of the question: we cannot deduce $\forall xFx$ from *all* instances of F.[1] The idea is thus to *assume* an instance and

1 It works only in a world (UD) with *countably* many instances of ϕ, and thus not generally.

generalise from that. But this requires *restrictions*: we need *arbitrary* constants in order to generalise universally. A constant is arbitrary if it is not associated with any reason for selecting it, or with any description or feature that we have already claimed, i.e. we make no assumptions about it at all. In that case, *c* would have special information: it needs to be nondescript and representative. Consider two points on the whiteboard:[2] we can draw *a* line between them and declare that *the* line is the shortest path between any two points, which holds not just for *this* real line on the whiteboard (the instance or exemplar), but for *every* line. We thus move from one *representative* instance to all instances of that kind.

The difficulty is that arbitrary constants are *constants just like any other*: they merely have a certain *function* within the proof. They are *used* in a special way. For instance, if '*a*' is already in play, and we chose *another* name '*b*', then it could be impossible to complete the proof. So, we need to select arbitrary constants carefully. For practical purposes, however, we can mark off arbitrary names with a macron ('$\bar{a}$').

Metalanguage	*QL*	
⋮	⋮	
$\dfrac{\phi\bar{a}}{\forall v[v/\bar{a}]}$	$\dfrac{{\sim}Fa \vee Ga}{\forall y({\sim}Fy \vee Gy)}$	$[\bar{a} = a]$ $[y = v]$

Restrictions	*Reason*
– $\phi\bar{a}$ does *not depend* on any formula that also contains *a*, e.g., in an undischarged assumption or in the premises	– *a* would not be arbitrary – excludes one-step arguments from one instance to all instances
– *a* does not occur in the quantified sentence we deduce	– this relates mainly to relations, such as $R\bar{a}\bar{a}$, where $\forall xRxa$ would be wrong (and $\forall xRxx$ would be right)

Note. In the Suppes' style deduction, we can simply check dependencies: the line to be generalised may not list a line that contains a formula that uses the same name as the line to be generalised (e.g., as an assumption, or in a premise).[3] Here is an example:[4]

1	(1)	$\forall x(Fx \vee Gx)$	P.
2	(2)	$\forall x(Gx \supset Hx)$	P. / $\forall x({\sim}Fx \supset Hx)$
1	(3)	$F\bar{a} \vee G\bar{a}$	1, ∀E [*a* is new, thus arbitrary]

2 See Tomassi, P. (1999). *Logic*. London: Routledge (p. 276).

3 This is why it is relevant to keep explicit note of the dependencies or (in Gentzen's style) to square-bracket assumptions once they are discharged.

4 For instance, 'everyone either is stupid or reads, and 'everyone who reads is a philosopher', so, 'everyone who is not stupid is a philosopher'.

2	(4)	$G\bar{a} \supset H\bar{a}$	2, ∀E [*do*.: new line]
3	(5)	$\sim Fa$	**A.**[thus '*a*' is no longer arbitrary]
1, **3**	(6)	Ga	3, ∨E [a quick and dirty version]
1, 2, **3**	(7)	Ha	4, 6 ⊃E
1, 2	(8)	$\sim F\bar{a} \supset H\bar{a}$	5–7, ⊃I [**A. discharged**]
1, 2	(9)	$\forall x(\sim Fx \supset Hx)$	8, ∀I □

∃Elim[5]

This is even more complicated, which is why most textbooks begin with background and informal explanations. The task is to deduce that a particular object in our UD is *F* from the fact that *some* (unknown) object is *F*. But how do we pick out the *right* one? $\exists xFx$ is really a possibly infinite disjunction (see Reader pp. 39, 53), so, if *some* object in our UD is *F*, we can say that $Fa \vee Fb \vee Fc \vee Fd, \ldots$. We cannot just say, $\exists xFx \vdash Fs$ (e.g., deduce that Socrates is funny from the fact that someone is funny). So, rather than derive, we *assume* (again) some *unknown* and *new* yet *specific* or *typical* instance. As this new constant is not really 'arbitrary' in the above sense, we mark it as '$\mathring{a}$'. (But the same idea is in play: it is the constant's function that makes it 'typical'.) The core idea is this: we select (or assume) an instance $\mathring{a}$ of an existentially quantified sentence *S* and use it to deduce another sentence in which $\mathring{a}$ does not appear anymore, and then we use *S* to prove that this sentence no longer depends on $\mathring{a}$, i.e. without assuming that $\mathring{a}$ is one of those things that have a certain predicate. This 'eliminates' $\exists v\phi$.

Metalanguage

$$\dfrac{\begin{array}{c} \\ \vdots \\ \exists v\phi \end{array} \qquad \begin{array}{c} [\phi\,[\mathring{a}/v]] \\ \vdots \\ \psi \end{array}}{\psi}$$

QL

$$\dfrac{\begin{array}{c} \\ \vdots \\ \exists x(\sim Fx \,\&\, Gx) \end{array} \qquad \begin{array}{c} \sim Fa \,\&\, Ga \\ \vdots \\ \exists yGy \end{array}}{\exists yGy} \qquad \begin{array}{l} [\mathring{a} = a] \\ \\ [y = v] \end{array}$$

Restrictions	*Reason*
– $\mathring{a}$ has not been used anywhere in the proof before (in the premises, in any undischarged assumption), and thus does not depend on any formula that also contains $\mathring{a}$	– *a* would not be a representative or typical instance – exclude one-step arguments from some instances to a particular instance
– $\mathring{a}$ does not occur in the sentence we deduce: ∃Elim can only be applied on *quantified* formulae	– *a* ceases to be used once we have shown what we intended; we repeat the last line to indicate we have proved *S* from $\exists v\phi$, rather than $\mathring{a}$.

5 This rule is similar to the ∨E rule in PL (Reader, p. 32), which allows us to remove the disjunction if we can prove some other proposition from both disjuncts.

Here is an example:

1	(1)	$\forall x(\sim Fx \supset Gx)$	P.
2	(2)	$\exists x \sim Fx$	P. / $\exists xGx$
3	(3)	$\sim F\mathring{a}$	**A.** [use $\mathring{a}$ *before* applying ∀Elim, …
1	(4)	$\sim F\mathring{a} \supset G\mathring{a}$	1, ∀E [… as we can use any constant]
1, 3	(5)	$G\mathring{a}$	3, 4 ⊃E
1, 3	(6)	$\exists xGx$	5, ∃I
1, 2	(7)	$\exists xGx$	2, 3, 6, ∃E [**A. discharged**] □

When there are multiple quantifiers, polyadic predicates, and modal operators, things become even more tangled; keeping track of *a*s, $\mathring{a}$s, and $\bar{a}$s is essential (and difficult). So here is an annotated example. We prove the validity of this argument: "Since everybody loves a lover, and Tom loves Sophie, everybody loves everybody."[6] Obviously, UD is the set of all persons. First, formalisation:

1	(1)	$\forall x(\exists yLxy \supset \forall zLzx)$	Premise	This reads, roughly, 'for each person (*x*), if there is some person (*y*)—to say 'other' person here would introduce complications of identity—that they love (i.e. if *x* is a lover), then everyone (*z*) loves this person (*x*)'.
2	(2)	Lts	Premise / $\forall x\forall yLxy$	The last step in the proof will be to apply ∀I twice, because there are two universal quantifiers in the conclusion.
2	(3)	$\exists yLty$	2, ∃I	So far, so good.
1	(4)	$\exists yLty \supset \forall zLzt$	1, ∀E [*t*/*x*]	Assume *t* because this constant is already in use, substitute *t* for *x* uniformly.
1, 2	(5)	$\forall zLzt$	3, 4 ⊃E	A simple application of ⊃E.
1, 2	(6)	$L\bar{a}t$	5, ∀E [$\bar{a}$/*z*]	We now apply ∀E, substituting the free(d) occurrence of *z* with $\bar{a}$, which stands for 'the one who loves Tom'.
1, 2	(7)	$\exists yL\bar{a}y$	6, ∃I [*y*/*t*]	In order to say that someone is a lover, or that there is someone that *u* loves, we apply ∃I, substituting *t* with *y*: we can use any variable—but *y* will appear in the conclusion.
1	(8)	$\exists yL\bar{a}y \supset \forall zLz\bar{a}$	1, ∀E [$\bar{a}$/*x*]	We instantiate the first premise again, but this time we use that representative lover $\bar{a}$ as the constant for *x*—any other constant makes the next step impossible.
1, 2	(9)	$\forall zLz\bar{a}$	7, 8 ⊃E	Another simple application of ⊃E.
1, 2	(10)	$Lc\bar{a}$	9, ∀E [*c*/*z*]	We apply ∀E again, and this time substituting *z* we use *c*, which stands for 'clandestine lover' (of that one who loves Tom). Note that *c* is arbitrary too. (Typographical constraints disallow macrons over cs.)
1, 2	(11)	$\forall yLcy$	10, ∀I [*y*/$\bar{a}$]	The generalisation works because the constant $\bar{a}$ we used was arbitrary (in its use); this reads, 'everybody is loved by a clandestine lover'. We choose *y* (instead of another *v* because of the conclusion.
1, 2	(12)	$\forall x\forall yLxy$	11, ∀I [*x*/*c*]	∀I applied again, replacing *c* with *x* and bind it with a universal quantifier. □

6 See Reader, pp. 54–5.