CEUR-WS.org/Vol-3875/ARQNL2024_paper8.pdf

A Proof-Theoretical Approach to Some Extensions of First Order Quantification

Loïc Allègre[1], Ophélie Lacroix[2] and Christian Retoré[1]

[1]LIRMM (Univ Montpellier & CNRS), Montpellier, France

[2]Resolve, Copenhagen, Danemark

Abstract

Generalised quantifiers, which include Henkin's branching quantifiers, have been introduced by Mostowski and Lindström and developed as a substantial topic application of logic, especially model theory, to linguistics with work by Barwise, Cooper, Keenan.

In this paper, we mainly study the proof theory of some non-standard quantifiers as second order formulae. Our first example is the usual pair of first order quantifiers (for all / there exists) when individuals are viewed as individual concepts handled by second order deductive rules. Our second example is the study of a second order translation of the simplest branching quantifier: "A member of each team and a member of each board of directors know each other", for which we propose a second order treatment.

Keywords

proof theory, second order logic, generalised quantifiers, branching quantifiers, individual concepts,

1. Generalisation of Usual First Order Quantification

Common first order quantifiers ∃ and ∀ have been formalised the way they are in standard mathematics by Frege [1] whose logical and philosophical view matches Hilbert desiderata regarding logical foundations of mathematics.[2, 3, 4, 5]

By that time, mathematicians and logicians were making little difference between the interpretations of quantifiers in *the* standard model and their proof rules — before the work of Skolem or Gödel, mathematicians and logicians made little distinction between syntax and semantics, they worked with an interpreted language, see e.g. [2] —— perhaps Hilbert was more demanding regarding quantifiers because he focused on foundations of mathematics [3].

Extensions of usual quantification have mainly been considered for faithfully modelling quantification modes that one finds in ordinary language, like numbers ("three students came to the party"), "most" ("most students came to the party"), percentages ("a third of the students came to the party"), vague quantifiers ("few students came to the party"), etc.

This gave rise to the theory of generalised quantifiers, initially intended for model theory [6, 7], that was intensively developed in connection with linguistics.[8, 9, 10]. In such a setting, the lexical item expressing a quantifier (say "most A are B") is viewed as a function with two

ARQNL 2024: Automated Reasoning in Quantified Non-Classical Logics, 1 July 2024, Nancy, France

loic.allegre@lirmm.fr (L. Allègre); ophelie.lacroix@resolve.tech (O. Lacroix); christian.retore@lirmm.fr (C. Retoré)
https://sites.google.com/site/ophelielacroixnlp/ (O. Lacroix); http://www.lirmm.fr/~retore (C. Retoré)
0000-0002-2401-9158 (C. Retoré)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

arguments that are predicates. This fits in well with the logico-functional view of quantification in Montague semantics [11]. Most (!) generalised quantifiers can be viewed as a second order construction. For instance the interpretation of a generalised quantifier depending on two unary predicates like "most" is interpreted as a second order construction, i.e. is true whenever the pairs of unary predicates it is applied to is a "legal" pair of subsets of the domain, namely a pair of sets (A, B) such that $|A \cap B| > |A \setminus B|$.

The paper is organised as follows. We first provide a reminder on second order logic, because this topic is not so common. Next, we present a second order view of usual first order quantification as second order quantification over individual concepts — and show the two formulations are proved to be equivalent provided the individual concepts are standard (i.e. they may not be empty, as opposed to some Kripke and Muskens variants). Then, after a quick presentation of generalised quantifiers, we study the simplest branching quantifier as a second order construction, and we propose direct rules for this quantifier.

2. A Reminder on Second Order Logic

We briefly remind the reader with basic facts about second order logic, following [12, Chapter 5], and one may also refer to the survey [13].

2.1. Language

A second-order language is based on a first-order language (the first three items in the list below), and extended with an infinitely enumerable set of predicate variables, each of them endowed with an arity (the last item in this list).

- an infinite enumerable set of first order (a.k.a. individual) variables x_i, with i in $\mathbb{N}$
- a set of constants c_i, with i in I (I is often enumerable, but this is not required);[1]
- an enumerable (or finite) set of predicate constants P_i^n with i in $\mathbb{N}$ each of them with an arity n (as in a first order language) – a predicate constant with arity 0 is a proposition;
- an enumerable set of predicate variables X_i^n i in $\mathbb{N}$ each of them with an arity n – a predicate variable with arity 0 is a propositional variable.

Second order formulae are defined "as expected": an atomic formula is $Z^n(u_1, \dots, u_n)$ with Z a predicate constant or a predicate variable of arity n and the u_i being n first order terms (here first order variables or first order constants since we have no functions). Les us call $\mathscr{A}$ the set of atomic formulae . Then the set of second order formulae $\mathscr{F}$ is defined by

$$\mathscr{F} =:: \mathscr{A} \mid \neg\mathscr{F} \mid \mathscr{F} \wedge \mathscr{F} \mid \mathscr{F} \vee \mathscr{F} \mid \mathscr{F} \rightarrow \mathscr{F} \mid \forall x_i \mathscr{F} \mid \exists x_i \mathscr{F} \mid \forall^n X_i^n \mathscr{F} \mid \exists^n X_i^n \mathscr{F}$$

where x_i stands for an individual variable while X_i^n stands for a predicate variable of arity n.

The formula $A \Leftrightarrow B$ is just a short-hand for $(A \rightarrow B) \wedge (B \rightarrow A)$.

[1] The first order language may also include an enumerable set of first order functions, but an n-ary function g in the language can replaced with an $n+1$-ary predicate G with an axiom that F is a function, and at second order there is a formula $F[X]$ saying that the $n+1$-ary predicate X corresponds to an n-ary function.

Although we shall not always write the ^{n} superscript in $\forall^n$ and $\exists^n$, beware that there are different pairs of second order quantifiers ($\exists^n/\forall^n$), one pair for each arity. The occurrences of the variable x_i or X^n_i are bound by the closest $\forall x, \exists x, \forall^n X^n_i, \exists^n X^n_i$ — if any — above them in the formula tree.

2.2. Proof Rules in Natural Deduction

We use natural deduction with standard rules as can be found in [12]. As we limit ourselves to classical logic, an extra principle is needed: *tertium non datur* ($A \vee \neg A$ for all A) or *reductio ad absurdum* (from a deduction with conclusion $\bot$ under hypothesis $\neg A$, conclude A), see e.g. [14]

The proof rules for second order quantifiers, namely the introduction and elimination rules of $\forall^n$ and $\exists^n$ are as expected, they are similar to the rules for first order quantifiers, *mutatis mutandi*:

$$\dfrac{\begin{array}{c}\vdots\\ \forall^n X^n_i\; T[X^n_i]\end{array}}{T[X^n_{i,k}(t^1_k,\dots,t^n_k) := \phi^n(t^1_k,\dots,t^n_k)]}\,(\forall^n)_E \qquad\qquad \dfrac{\begin{array}{c}\vdots\\ T[X^n_i]\end{array}}{\forall^n X^n_i\; T[X^n_i]}\,(\forall^n)_I$$

$$\dfrac{\exists^n X^n_i\; T[X^n_i] \qquad \begin{array}{c}\Big[T[X^n_i]\Big]^k\\ \vdots\\ \psi\end{array}}{\psi}\,(\exists^n)^k_E \qquad\qquad \dfrac{\begin{array}{c}\vdots\\ T[X^n_{i,k}(t^1_k,\dots,t^n_k) := \phi^n(t^1_k,\dots,t^n_k)]\end{array}}{\exists^n X^n_i T[X^n_i]}\,(\exists^n)_I$$

where

1. $T[X^n_i]$ stands for a formula in which the predicate variable X^n_i may occur (but that is not mandatory, as for first order quantification).
2. There should be no free occurrence of X^n_i in the hypotheses of the introduction rule $(\forall^n)_I$ nor in the elimination rule $(\exists^n)^k_E$ — as in the first order $\forall_I$ and $\exists_E$ introduction rules.
3. The obscure notation[2] $T[X^n_{i,k}(t^1_k,\dots,t^n_k) := \phi^n(t^1_k,\dots,t^n_k)]$ requires some explanation. This formula stands for the formula obtained by replacing
   - the k^{th} occurrence $X^n_{i,k}$ of X^n_i which is applied to n terms $(t^1_k,\dots,t^n_k)$
   - with a formula with n free variables applied to the very same terms $(t^1_k,\dots,t^n_k)$

   with the requirement that no originally free variable in $(t^1_k,\dots,t^n_k)$ becomes bound after the application of ϕ to $(t^1_k,\dots,t^n_k)$.
   Here is an example: let $\phi(x,y) = P(x,y) \wedge Q(y,a)$, let $T[X^2_1] = X^2_{1,1}(z,a) \wedge X^2_{1,2}(a,b)$ – mind the second subscript of $X^2_{1,\bullet}$ which indicates the occurrence number (there are two occurrences of the predicate variable X^2_1 in $T[X^2_1]$. Then $T[X^2_{1,k} := \phi(t^1_k,\dots,t^n_k)]$ is

[2] This point is often under explained in the literature.

$(P(x,y) \wedge Q(y,a))[x := z; y := a] \wedge (P(x,y) \wedge Q(y,a))[x := a; y := b]$ that is $(P(z,a) \wedge Q(a,a)) \wedge (P(a,b) \wedge Q(b,a))$. From the definition and the example, it is unsurprising that second order unification is undecidable [15].

4. In the rule $(\exists^n)_E$ the expression $\left[T[X_i^n]\right]^k$ indicates that the hypothesis $T[X_i^n]$ has been cancelled during the $(\exists^n)_E^k$ number k.

Some remarks:

1. this proof system can derive the comprehension axiom:

$$\exists X^n \forall x_1 \dots x_n \left[\varphi(x_1,\dots,x_n) \leftrightarrow X^n(x_1,\dots,x_n)\right]$$

2. equality can be defined à la Leibnitz: $x = y \,:\, \forall^1 X^1 \left[X^1(x) \to X^1(y)\right]$ (because of negation there is no need to use ⇔ in this definition)
3. being equal to x is a property $E_x(y) \,:\, \forall^1 X^1 \left[X^1(x) \to X^1(y)\right]$.
4. the Dedekind finiteness, “any injective function is surjective” is definable:
$\forall^2 X^2 \quad ((\forall x \forall y \forall z (X(x,y) \wedge X(x,z) \to y = z)) \wedge (\forall x \forall y \forall z (X(y,x) \wedge X(z,x) \to y = z)))$
$\to (\forall w \exists u\, X(u,w))$

Finally, using implication →, first order $\forall$, and propositional second order $\forall^0$ one can define false, $\perp$, negation $\neg$, the propositional connective $\wedge$, $\vee$, first order existential quantification $\exists$. Adding $\forall^n$ to →, $\forall$, one can also define $\exists^n$. Thus, the expressive power of second order propositional quantifier $\forall^0$ is impressive.

2.3. Standard and Non-standard Models, Completeness

We here follow [16, 13, 12].

A second order model consists in a first order model, i.e. with a domain D, endowed with a set of sets of tuples of length n for each $n \in \mathbb{N}$ in order to interpret predicate variables of arity n: they may vary in a fixed subset A^n of $\mathscr{P}(D^n)$ which is not necessarily the full powerset $\mathscr{P}(D^n)$; for this structure to define a model, it must enjoy the comprehension axiom scheme $\exists^n X^n \forall x_1 \cdots \forall x_n [\phi(x_1,\dots,x_n) \leftrightarrow X^n(x_1,\dots,x_n)]$ where the n-ary predicate variable X^n does not appear in ϕ — in other words the subsets of A^n must include the interpretations of the formulae with n free variables. The comprehension axiom scheme is derivable from the existential introduction rule given above.

By definition, a second-order model is said to be *standard* (or *full*) whenever the subset A^n is $\mathscr{P}(D^n)$ for any arity n. Standard models satisfy the comprehension scheme. They do match intuition: a predicate variable of arity n varies in all possible subsets of D^n. However, neither completeness nor compactness hold when only standard models are considered — indeed second order logic can express the finiteness of the domain D, see *e.g.* [12].

Second order logic may be encoded in first order logic: predicate variables X_i^n are viewed as individual constants interpreted in a domain (that also contains standard individuals), and some additional predicate constants of arity $n+1$ are needed to mimic the application of a predicate variable of arity n to n terms. Then a second order formula is provable within second order logic whenever its first order translation is provable in first order logic. So applying first order completeness theorem one gets that a second order formula is provable in second order logic if and only if it is true in all second order models (including the non standard ones). As a consequence of completeness, compactness holds, so one can have a model with at least n elements for each integer n, which is Dedekind finite.

3. A Second Order View of First Order Quantification: Quantifying Over Individual Concepts

3.1. Individual Concepts

Individual concepts view an individual as a formula $\phi[x]$ with a single free variable x such that there is a single individual satisfying the formula $\phi[x]$.[3] This can be said in second order logic: a formula $\phi[x]$ with a single free variable x is said to be an individual concept whenever there is at most one individual x such that $\phi[x]$ and at least one x such that $\phi[x]$ — so it makes exactly one x such that $\phi[x]$. That the concept ϕ is an *individual concept* can actually be expressed in second order logic, where the "=" symbol refers to usual equality (with its usual rules: reflexivity, symmetry, transivity, substition etc. noted "=" in the proofs):[4]

$$C(\phi) := (\forall x \forall y[\phi(x) \land \phi(y) \to x = y]) \land \exists z\, \phi(z)$$

Individual concepts are close to Montague semantics and Leibnitz identity (where an individual is identified with the set of all properties it enjoys) [18, 19, 11]. For reasons like possible worlds semantics, see e.g. the discussion in [20, chapter 4], some logicians consider a variant of individual concepts that are possibly empty. This may look strange but if you think individual concepts are some kind of proper name that are part of the logical language, it is hard to tell what a proper name refers to before the reference actually exists. For instance, the individual concept Gödel(x) has no reference in Egyptian times. Hence Kripke in the 70s dropped the existence condition from individual concepts (see [21] for a recent account of those ideas), thus obtaining a formula with a lower logical complexity profile:

$$C_\epsilon(\phi) := \forall x \forall y[\phi(x) \land \phi(y) \to x = y]$$

So $C(\phi)$ can be defined as $C(\phi) := C_\epsilon(\phi) \land \exists z\, \phi(z)$.

3.2. First Order Universal Quantification as Universal Quantification Over Non-empty Individual Concepts

The simplest quantifiers one can try to view as a second order construction are clearly the usual first order quantifiers $\forall, \exists$. So let us compare the second order quantification over individual concepts to usual quantification.

[3] This notion of concept is somehow related to concepts in description logics, and the individual concepts that we use correspond to individual names cf. e.g. [17].

[4] The ":=" symbol denotes an abbreviation (or a definition) of a formula, and in proofs, the replacement of an abbreviation by its expansion or vice-versa.

Proposition 1. *First order universal quantification and second order quantification over individual concepts are equivalent:*

1. *given a property $\varphi(X)$ of individual concepts, the following equivalence holds: $\forall x\, \varphi^{\downarrow}(x) \Leftrightarrow \forall X(C(X) \to \varphi(X))$ where $\varphi^{\downarrow}(x) := \exists X(C(X) \land X(x) \land \varphi(X))$.*
2. *given a property $\psi(x)$ of individuals, the following equivalence holds: $\forall x\ \psi(x) \Leftrightarrow \forall X\big(C(X) \to \psi^{\uparrow}(X)\big)$ where $\psi^{\uparrow}(X) := \exists x(X(x) \land \psi(x))$.*

Proof. Let us first observe that there is a simple formal proof without assumption that $C(E_x)$ i.e. that "being equal to x" (cf. section 2 item 3) is an individual concept, $E_x(y) : y = x : \forall^1 X^1\ (X^1(x) \to X^1(y))$, and let us call this proof δ, because we are going to use it several times:

$$\delta :$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[y=x\land z=x]}{y=x}\,\land E \qquad \dfrac{\dfrac{[y=x\land z=x]}{z=x}\,\land E}{x=z}\,=}{y=z}\,=}{y=x\land z=x\to y=z}\,\to I}{\forall z(y=x\land z=x\to y=z)}\,\forall^1 I}{\forall y\forall z(y=x\land z=x\to y=z)}\,\forall^1 I}{C(E_x)}\,:=$$

1. a) Assuming $\forall x\, \varphi^{\downarrow}(x)$ one can prove $\forall X(C(X) \to \varphi(X))$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\forall x\varphi^{\downarrow}(x)] \qquad \dfrac{\dfrac{\dfrac{[C(X)]}{\forall x\forall y(X(x)\land X(y)\to x=y)\land\exists xX(x)}}{\exists xX(x)}\,\land E \qquad [X(x)]}{X(x)}\,\exists^1 E}{\forall x\varphi^{\downarrow}(x)\land X(x)}\,\land I}{\varphi^{\downarrow}(x)}}{\exists YC(Y)\land Y(x)\land\varphi(Y)}\,:= \qquad [C(Y)\land Y(x)\land\varphi(Y)]}{C(Y)\land Y(x)\land\varphi(Y)}\,\exists^2 E}{Y(x)\land\varphi(Y)}\,\land E}{\varphi(Y)}\,\land E}{\varphi(X)}\,:=}{C(X)\to\varphi(X)}\,\to I}{\forall X(C(X)\to\varphi(X)}\,\forall^2 E$$

b) Assuming $\forall X(C(X) \to \varphi(X))$ one can prove $\forall x\, \varphi^{\downarrow}(x)$. In the proof below, we use δ the proof that $E_x(\_) = "\_ = x"$ (that is, "being equal to x") is an individual concept.

$$
\begin{prooftree}
\AxiomC{δ}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$C(E_x)$}
\AxiomC{$x=x$}
\UnaryInfC{$E_x(x)$}
\AxiomC{δ}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$C(E_x)$}
\AxiomC{$[\forall X(C(X)\to\varphi(X))]$}
\RightLabel{$\forall E$}
\UnaryInfC{$C(E_x)\to\varphi(E_x)$}
\RightLabel{$\to E$}
\BinaryInfC{$\varphi(E_x)$}
\RightLabel{$\wedge I$}
\BinaryInfC{$E_x(x)\wedge\varphi(E_x)$}
\RightLabel{$\wedge I$}
\BinaryInfC{$C(E_x)\wedge E_x(x)\wedge\varphi(E_x)$}
\RightLabel{$\exists^2 I$}
\UnaryInfC{$\exists X(C(X)\wedge X(x)\wedge\varphi(X))$}
\RightLabel{$:=$}
\UnaryInfC{$\varphi^{\downarrow}(x)$}
\RightLabel{$\forall^1 E$}
\UnaryInfC{$\forall x\varphi^{\downarrow}(x)$}
\end{prooftree}
$$

2. a) Assuming $\forall x\, \psi(x)$ one can prove $\forall X\big(C(X) \to \psi^{\uparrow}(X)\big)$

$$
\begin{prooftree}
\AxiomC{$[C(X)]$}
\RightLabel{$:=$}
\UnaryInfC{$\forall x\forall y(X(x)\wedge X(y)\to x=y)\wedge\exists x\, X(x)$}
\RightLabel{$\wedge E$}
\UnaryInfC{$\exists x X(x)$}
\AxiomC{$[X(x)]$}
\RightLabel{$\exists^1 E$}
\BinaryInfC{$X(x)$}
\AxiomC{$[\forall x\psi(x)]$}
\RightLabel{$\forall^1 E$}
\UnaryInfC{$\psi(x)$}
\RightLabel{$\wedge I$}
\BinaryInfC{$X(x)\wedge\psi(x)$}
\RightLabel{$\exists^1 I$}
\UnaryInfC{$\exists x(X(x)\wedge\psi(x))$}
\RightLabel{$:=$}
\UnaryInfC{$\psi^{\uparrow}(X)$}
\RightLabel{$\to I$}
\UnaryInfC{$C(X)\to\psi^{\uparrow}(X)$}
\RightLabel{$\forall^2 I$}
\UnaryInfC{$\forall X(C(X)\to\psi^{\uparrow}(X)$}
\end{prooftree}
$$

b) Finally assuming $\forall X\big(C(X) \to \psi^{\uparrow}(X)\big)$ one can prove: $\forall x\, \psi(x)$

$$
\begin{prooftree}
\AxiomC{δ}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$C(E_x)$}
\AxiomC{$[\forall X(C(X)\to\psi^{\uparrow}(X)]$}
\RightLabel{$\forall^2 E$}
\UnaryInfC{$C(E_x)\to\psi^{\uparrow}(E_x)$}
\RightLabel{$\to E$}
\BinaryInfC{$\psi^{\uparrow}(E_x)$}
\RightLabel{$:=$}
\UnaryInfC{$\exists y E_x(y)\wedge\psi(y)$}
\RightLabel{$\exists^1 E$}
\UnaryInfC{$E_x(y)\wedge\psi(y)$}
\RightLabel{$:=$}
\UnaryInfC{$y=x\wedge\psi(y)$}
\RightLabel{$\wedge E$}
\UnaryInfC{$\psi(x)$}
\RightLabel{$\forall^1 I$}
\UnaryInfC{$\forall x\psi(x)$}
\end{prooftree}
$$

□

3.3. First Order Existential Quantification as Existential Quantification Over Non-empty Individual Concepts

As for the universal quantification, we have:

Proposition 2. *First order existential quantification and second order quantification over individual concepts are equivalent:*

1. *when ϕ is a property of individual concepts, one has $\exists x\, \varphi^{\downarrow}(x) \Leftrightarrow \exists X(C(X) \land \varphi(X))$ where $\varphi^{\downarrow}(x) := \exists X(C(X) \land X(x) \land \varphi(X))$.*
2. *when ψ is a property of individuals, one has $\exists x\, \psi(x) \Leftrightarrow \exists X\big(C(X) \land \psi^{\uparrow}(X)\big)$ where $\psi^{\uparrow}(X) := \exists x(X(x) \land \psi(x))$*

Proof. At point 2.a) we will also use the proof δ of $C(E_x)$ from the proof of proposition 1.

1. $\exists x\, \varphi^{\downarrow}(x) \Leftrightarrow \exists X(C(X) \land \varphi(X))$ where $\varphi^{\downarrow}(x) := \exists X(C(X) \land X(x) \land \varphi(X)$
   a) Let us prove $\exists X(C(X) \land \varphi(X))$ under the assumption $\exists x\, \varphi^{\downarrow}(x)$.

$$\dfrac{[\exists x \varphi^{\downarrow}(x)] \qquad \dfrac{\dfrac{[\varphi^{\downarrow}(x)]}{\exists X(C(X) \land X(x) \land \varphi(X))} := \qquad \dfrac{\genfrac{}{}{2pt}{0}{[C(X) \land X(x) \land \varphi(X)]}{C(X)} \land E \qquad \genfrac{}{}{2pt}{0}{[C(X) \land X(x) \land \varphi(X)]}{\varphi(X)} \land E}{C(X) \land \varphi(X)} \exists^2 I}{\exists X(C(X) \land \varphi(X))} \exists^2 E}{\exists X(C(X) \land \varphi(X))} \exists^1 E$$

   b) Now let us prove $\exists x\, \varphi^{\downarrow}(x)$ under the assumption $\exists X(C(X) \land \varphi(X))$. We first need α, β, γ i.e. the three following proofs :

α :

$$\dfrac{[\exists X(C(X) \land X(x) \land \varphi(X))] \qquad \dfrac{[C(X) \land \varphi(X)]}{C(X)} \land E}{C(X)} \exists^2 E$$

β :

$$\dfrac{[\exists X(C(X) \land X(x) \land \varphi(X))] \qquad \dfrac{[C(X) \land \varphi(X)]}{\varphi(X)} \land E}{\varphi(X)} \exists^2 E$$

γ :

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\alpha\\ \vdots\\ C(X)\end{matrix}}{\forall x\forall y(X(x)\wedge Y(x)\to x=y)\wedge\exists xX(x)}\;:=}{\exists xX(x)}\;\wedge E \qquad [X(x)]}{X(x)}$$

and the proof we are looking for is:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\begin{matrix}\alpha\\ \vdots\\ C(X)\end{matrix}\quad\begin{matrix}\gamma\\ \vdots\\ X(x)\end{matrix}}{C(X)\wedge X(x)}\;\wedge I \quad \begin{matrix}\beta\\ \vdots\\ \varphi(X)\end{matrix}}{C(X)\wedge X(x)\wedge\varphi(X)}\;\wedge I}{\exists X(C(X)\wedge X(x)\wedge\varphi(X))}\;\exists^2 I}{\varphi^{\downarrow}(x)}\;:=}{\exists x\varphi^{\downarrow}(x)}\;\exists^1 I$$

2. $\exists x\,\psi(x) \Leftrightarrow \exists X\big(C(X)\wedge\psi^{\uparrow}(X)\big)$ where $\psi^{\uparrow}(X) := \exists x(X(x)\wedge\psi(x))$

   a) Let us prove $\exists X\big(C(X)\wedge\psi^{\uparrow}(X)\big)$ under the assumption $\exists x\,\psi(x)$ – δ is the proof of $C(E_x)$ defined in the proof of proposition 1.

$$\dfrac{\dfrac{\begin{matrix}\delta\\ \vdots\\ C(E_x)\end{matrix}\qquad \dfrac{\dfrac{\begin{matrix}\delta\\ \vdots\\ C(E_x)\end{matrix}\quad \dfrac{\dfrac{\dfrac{x=x}{E_x(x)}\quad \dfrac{[\exists x\psi(x)]\quad[\psi(x)]}{\psi(x)}\;\exists^1 E}{E_x(x)\wedge\psi(x)}\;\wedge I}{\exists y(E_x(y)\wedge\psi(y))}\;\exists^1 I}{C(E_x)\wedge\exists y(E_x(y)\wedge\psi(y))}\;\wedge I}{\psi^{\uparrow}(E_x)}\;:=}{C(E_x)\wedge\psi^{\uparrow}(E_x)}\;\wedge I}{\exists X(C(X)\wedge\psi^{\uparrow}(X)}\;\exists^2 I$$

b) Let us prove $\exists x\,\psi(x)$ under the assumption $\exists X\big(C(X)\wedge\psi^{\uparrow}(X)\big)$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\exists X(C(X)\wedge\psi^{\uparrow}(X))] \quad [C(X)\wedge\psi^{\uparrow}(x)]}{C(X)\wedge\psi^{\uparrow}(X)}\,\exists^2E}{\psi^{\uparrow}(X)}\,\wedge E}{\exists x(X(x)\wedge\psi(x))}\,:= \qquad\qquad \dfrac{[X(x)\wedge\psi(x)]}{\psi(x)}\,\wedge E}{\psi(x)}\,\exists^1E}{\exists x\psi(x)}\,\exists^1I$$

□

3.4. Dealing With Possibly Empty Individual Concepts

When individual concepts are possibly empty the second order formula expressing that X is an individual concept is $C_\epsilon(X) = \forall x\forall y(Xx\wedge Xy\rightarrow x = y)$ — the $\exists x X(x)$ left out of our initial definition of individual concepts.

Regarding universal quantification, $\forall X(C_\epsilon(X)\rightarrow\varphi(X))$ still entails $\forall x\,\varphi^{\downarrow}(x)$, but $\forall x\,\varphi^{\downarrow}(x)$ does not entail $\forall X(C_\epsilon(X)\rightarrow\varphi(X))$ anymore. This is logical: when all individual concepts have a property be they empty or not, all individuals enjoy the corresponding first order property. The converse does not hold: when all individuals enjoy a property, all the non empty concepts enjoy the property, but why should the empty individual concept enjoy this property as well?

Of course regarding existential quantification, that's the opposite. $\exists x\,\varphi(x)$ entails $\exists X(C_\epsilon(X)\wedge\varphi(X)))$ but $\exists X(C_\epsilon(X)\wedge\varphi(X))$ does not entail $\exists x\,\varphi(x)$. When an individual enjoys a property, so does the corresponding individual concept. But when a possibly empty individual concept enjoys a property, it does not entail that an individual enjoys this property, because this individual concept might be an empty individual concept.

So the second order view of usual quantification does not fit in well with possibly empty individual concepts.

4. A Reminder on Generalised and Branching Quantifiers

This reminder mainly relies on the presentation given by Peters and Westerståhl [22], one may also consult the survey [10]. Generalised quantifiers, initially introduced by Mostowski [6] and further developed by Lindström [7] are a generalisation of standard universal and existential quantification. Roughly speaking, generalised quantifiers view quantifiers as relations over relations (or tuples of relations) — those relations are relations on the domain (a.k.a universe, model) of an interpretation: thus, quantifiers are viewed as second-order concepts.

4.1. Generalised Quantifiers

Given k integers $n_1, ..., n_k$, a quantifier $\mathcal{Q}$ of type $\langle n_1, ..., n_k\rangle$ can be viewed as a function endowing each domain M with a k-ary relation Q_M such that if $(R_1, ..., R_k)\in Q_M$, then for all i in between 1 and k, R_i is a n_i-ary relation over elements of M. Let us give some examples.

The usual quantifiers $\forall$ and $\exists$ can be then expressed as simple type $\langle 1\rangle$ quantifiers : $\exists_M = \{A \subseteq M,\ A \neq \emptyset\}$ and $\forall_M = \{A \subseteq M,\ A = M\}$. Thus, the existential quantifier is in every domain the unary relation which holds true for all non-empty predicates, and the universal quantifier is the relation which holds true only for the whole domain M.

Some generalised quantifiers have an equivalent formulation in usual first-order logic, such as the quantifier "at least two": $(\exists_{\geq 2})_M = \{A \subseteq M,\ |A| \geq 2\}$ which can be expressed with the following first-order formula: $\exists x \exists y[x \neq y]$. However this is not always the case. Take for example the type $\langle 1, 1\rangle$ quantifier expressing that most A are B: $\mathbf{Most}_M(A, B) \iff |A \cap B| > |A - B|$ which notably cannot be expressed as a first-order formula.

It is worth noting that universal and existential quantification on individual concepts as we presented in Section 3 can also be formulated in terms of generalised quantifiers. To say that all (resp. some) individual concepts satisfy a property φ is in fact a second-order statement about the predicates C ("to be an individual concept") and φ. Hence the second order view of first order quantification that we presented in the previous section can be expressed as generalised quantifiers $\forall_C$ and $\exists_C$ with type $\langle 1, 1\rangle$:

$$\forall_C(C, \varphi) \iff C \subset \varphi \qquad \exists_C(C, \varphi) \iff C \cap \varphi \neq \emptyset$$

4.2. Branching Quantifiers

Among generalised quantifiers, branching quantifiers are of particular interest, both for logic and linguistics. Initially introduced by Henkin [23] and much later on studied by Hintikka [24] — independently of generalised quantifiers — branching quantification is a generalisation of classical quantification that allows the expression of independence between some existentially quantified variable and some previously universally quantified variables. This cannot be expressed within usual quantification because quantifiers are supposed to be linearly ordered. The simplest example of such a non-first-order quantifier is the following Henkin quantifier where, as the notation suggests, x' only depends on x, while y', only depends on y.

$$\begin{array}{l}\forall x \exists x' \\ \forall y \exists y'\end{array} \Big> F(x, y, x', y')$$

As proven by Ehrenfeucht (in Henkin [23]), this construction has no first-order equivalent. Notably, it cannot be expressed with a linear quantifier prefix such as $\forall x \exists x' \forall y \exists y'$ or $\forall x \forall y \exists x' \exists y'$, since there would be unwanted dependencies between x' and y, and y' and x.

Although not initially introduced as such, branching quantifiers can in fact be seen as specific generalised quantifiers. Indeed, the (in)dependencies between variables can be expressed using Skolem functions, e.g. the Henkin quantifier above can be written as follows:

$$\exists f \exists g \forall x \forall y\, F(x, f(x), y, g(y))$$

This in turn allows us to translate it as a generalised quantifier, for example here as the type $\langle 4\rangle$ quantifier:

$$H_M = \{R \subseteq M^4 \mid \exists f \exists g \forall x \forall y\, (x, f(x), y, g(y)) \in R\}$$

5. Second Order Proof Rules for Branching Quantifiers

Branching Quantifiers as Second Order Formulae In this part, we focus on the expression of branching quantifiers as second-order constructions. Such quantifiers can be quite complex, so we limit ourselves to studying the simplest branching quantifier. Our main object of study is the typical branching constructions found in natural language in the so-called Hintikka sentences, such as :

(H) *A member of each team and a member of each board of directors know each other*

The branching-quantifier reading of the above English sentence can be formulated within second-order logic:[5]

$$\begin{array}{l}\forall x \exists x' \\ \\ \forall y \exists y'\end{array} \Big> T(x) \wedge B(y) \rightarrow M(x, x') \wedge M(y, y') \wedge K(x', y')$$

As mentioned earlier, this formula can be expressed as a second order formula with existential quantification over functions:

$$(\mathsf{Hfun}) : \exists f \exists g \forall x \forall y \, [T(x) \wedge B(y)] \rightarrow K(f(x), g(y))$$

Natural Deduction Rules With Binary Predicates This formulation of the Henkin quantifier as a second-order formula with quantification over functions is however not fully satisfactory, for it actually provides a stronger effect than needed: defining f and g as functions implies the unicity of $f(x)$ and $g(y)$ for any given x and y, while the original formula with a branching quantifier only requires that there exists one (possibly more) x' for each x and y' for each y. Thus f and g need not be functions, but only need be non-empty binary predicates — as always with Skolem functions, the choice of $f(x)$ for each x is part of the interpretation of the function symbol.

Therefore, we propose another second-order representation of this reading of the sentence using quantification over predicates instead of quantification over functions:

$$\begin{aligned}(\mathsf{Hpred}) : &\exists F \exists G [\forall x \exists x' T(x) \rightarrow F(x, x')] \wedge [\forall y \exists y' B(y) \rightarrow G(y, y')] \\ &\wedge [\forall x \forall x' \forall y \forall y' [T(x) \wedge B(y) \wedge F(x, x') \wedge G(y, y')] \rightarrow K(x', y')]\end{aligned}$$

This formula simply replaces each of the two functions f and g of $(H fun)$ above with the binary predicates F and G. These two predicates act intuitively as relations that select suitable x' and y', since all we need to ensure is that whenever x' is a valid representative for x (and y' for y), then x' and y' know each other. The binary predicates F and G are required to relate each possible value of their first argument which ought to be in the proper set/predicate (T for x, B for

[5] There is also a first-order reading of this sentence, which the 'each other' (perhaps) makes less perceptible, and which can be expressed within first-order logic: $[\forall x \exists x' \forall y \exists y' \quad T(x) \wedge B(y) \rightarrow M(x, x') \wedge M(y, y') \wedge K(x', y')] \wedge [\forall y \exists y' \forall x \exists x' \quad T(x) \wedge B(y) \rightarrow M(x, x') \wedge M(y, y') \wedge K(x', y')]$. According to Szymanik [25], in two-thirds of cases, the first-order reading is preferred to the branching quantifier reading.

y) to at least one value of their second argument.[6] There is nevertheless a difference between using function as in (Hfun) and (Hpred): in (Hpred) there may well exist several values of x' (resp. y') such that $F(x, x')$ (resp. $G(y, y')$), there is no need to chose one as opposed to (Hfun), where one is explicitly chosen.

The natural deduction rules for second-order logic from section 2.2 give us the introduction and elimination rules for the branching Henkin quantifier. For the sake of readability, let us write from here on:

$$\Phi(F, G, x, x', y, y') = [T(x) \wedge B(y) \wedge F(x, x') \wedge G(y, y')] \rightarrow K(x', y')$$

and

$$\begin{aligned}\Psi(F, G) = [\forall x \exists x' T(x) \rightarrow F(x, x')] &\wedge [\forall y \exists y' B(y) \rightarrow G(y, y')] \\ &\wedge [\forall x, \forall x' \forall y \forall y' \Phi(F, G, x, x', y, y')]\end{aligned}$$

The introduction rule is quite straightforward:

$$\dfrac{\varphi(x, t) \qquad \psi(y, u) \qquad \Phi(\varphi, \psi, x, x', y, y')}{\exists F \exists G[\forall x \exists x' T(x) \rightarrow F(x, x')] \wedge [\forall y \exists y' B(y) \rightarrow G(y, y')] \wedge [\forall x, \forall x' \forall y \forall y' \Phi(F, G, x, x', y, y')]}\, H_I$$

where φ is a formula with free variables exactly x, x', ψ a formula with free variables exactly y, y', and t, u are terms.

The elimination rule, however, is more complicated due to the use of two second-order eliminations of $\exists^2$:

$$\dfrac{\exists F \exists G \Psi(F, G) \qquad \dfrac{[\exists G \Psi(A, G)]^{(2)} \qquad \begin{array}{c}[\Psi(A, B)]^{(1)} \\ \vdots \\ \varphi\end{array}}{\varphi}\, \exists^2 E^{(1)}}{\varphi}\, \exists^2 E^{(2)}$$

in which A and B must not appear free in φ.

Let us write as above $H(x, x', y, y')$ the branching quantifier that binds the two universally quantified variables x, y and the two existentially quantified variables x', y' e.g. the example above may be written $H(x, x', y, y')\ \Phi(F, G, x, x', y, y')$.

Let $\mathscr{H}$ be the set of closed formulae that can be written with this quantifier and the two usual first order quantifiers $\exists x$ and $\forall y$.

In the near future, we intend to determine whether our direct rules can be used to derive all the formulae in $\mathscr{H}$ that can be derived with the usual rules for second and first quantifiers. It seems plausible to us, because cut-elimination holds for second order logic. [26] However, we are not yet fully certain, and this is one aspect that we intend to clarify in the near future.

[6] Similarly, we could ask that x' and y' are in the proper set/predicate (T for x', B for y') but it is less important. Thus we do not add this precision, which is not needed — unlike the restriction to x and y — and makes the formulae, which are already long enough, considerably longer: Hpred′ : $\exists F \exists G[\forall x \exists x' T(x) \rightarrow F(x, x') \wedge T(x')] \wedge [\forall y \exists y' B(y) \rightarrow G(y, y') \wedge B(y')] \wedge [\forall x \forall x' \forall y \forall y' T(x) \wedge T(x') \wedge B(y) \wedge B(y') \wedge F(x, x') \wedge G(y, y') \rightarrow K(x', y')]$

6. Conclusion

This ongoing work deals with the proof rules of extensions of first order quantification viewed as second order logic constructions which do not need the full expressive power (and complexity) of second order logic.

We first described usual first order quantifiers within the second order logic as quantification over individual concepts and obtained that this description works provided individual concepts are asked to be non empty — unsurprisingly it does not work without this restriction.

Thereafter we focused on the non first order reading of the simplest branching quantifier that one finds in sentences like: *"A member of each team and a member of each board of directors know each other"*. Regarding this (reading of this) quantifier, we proposed a definition of it within second order logic and provided direct introduction and elimination rules for this complex quantifier which is often only described in model-theoretic terms. Later on we shall prove that the complete set of second order rules does not derive more sentences with those connectives than our direct rules.

While we were working on the final version of this article, we realised that Matthias Baaz and Anela Lolic [27] proposed an analytic calculus for Henkin quantifiers. In their paper, Henkin quantifiers are viewed as the existence of functions (i.e. as a particular form of second order quantification), and they proved cut-elimination for this calculus. It is too late for this paper of ours to examine how their account of Henkin quantifiers differs from ours, but this will be be our first aim in continuing our work. A first remark is that (1) : $\exists F\, \forall u\, \exists v\, F(u,v)$ seems slightly weaker than (2) : $\exists f\, F(u, f(u))$: in order to derive (2) from 1, it is necessary to utilise some form of the axiom of choice. Another difference, a very small one actually, is that our rules are natural deduction rules (introduction/elimination rules) and not sequent calculus: sequent-calculus right-rules correspond to natural-deduction introduction-rules but sequent-calculus left-rules do not correspond to natural-deduction elimination-rules.

By way of conclusion, let us mention a prospect that has opened up to us recently. The epsilon calculus [28, 29] may express readings that are close to branching quantifier readings, with epsilon formulas that have no equivalent in first or higher order logic[7]. Indeed, the subnectors epsilon and tau express quantification with some scope ambiguities (under-specification)[8]. However, it is presently too early to say anything definite about this question.

Acknowledgments

We would like to express our gratitude to the anonymous reviewers and to the chairs of ARQNL2024 for their valuable feedback, which has been instrumental in helping us to enhance this article. Furthermore, we would like to extend our gratitude to the programme chairs who accepted some late modifications.

[7] The formula $A(\varepsilon_v B(v))$ is not equivalent to any first order formula — although it is equivalent to $\exists v\, A(v) \wedge B(v)$ when additionally $\exists v\, B(v) \equiv B(\varepsilon_v B(v))$ holds.

[8] As an example of under-specification or scope ambiguity in the ε-calculus, a formula like $G(\varepsilon_u A(u), \tau_w B(w))$ which has no equivalent in first order logic, has some logical relation, depending on whether A and B are $\emptyset$ or D, with the two first-order formulas $\forall u \exists w\, A(u) \implies (B(w) \wedge G(u,w))$ and $\exists w \forall u\, B(w) \wedge (A(u) \implies G(u,w))$.

References

[1] G. Frege, Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens, Verlag von Louis Nebert, Halle, 1879.

[2] W. Kneale, M. Kneale, The development of logic, 3rd ed., Oxford University Press, 1986.

[3] W. D. Goldfarb, Logic in the twenties: The nature of the quantifier, J. Symb. Log. 44 (1979) 351–368. URL: https://doi.org/10.2307/2273128. doi:10.2307/2273128.

[4] D. Hilbert, P. Bernays, Grundlagen der Mathematik. Bd. 1., Berlin: Julius Springer. XII, 471 S., 1934. Traduction française de F. Gaillard et M. Guillaume, L'Harmattan, 2001.

[5] D. Hilbert, P. Bernays, Grundlagen der Mathematik. Bd. 2., Springer, 1939. Traduction française de F. Gaillard, E. Guillaume et M. Guillaume, L'Harmattan, 2001.

[6] A. Mostowski, On a generalization of quantifiers, Fundamenta Mathematicae 44 (1957) 12–36.

[7] P. Lindström, First order predicate logic with generalized quantifiers, Theoria 32 (1966) 186–195.

[8] J. Barwise, R. Cooper, Generalized quantifiers and natural language, Linguistics and Philosophy 4 (1981) 159–219. doi:10.1007/BF00350139.

[9] E. Keenan, J. Stavi, A semantic characterization of natural language determiners, Linguistic and Philosophy 9 (1986) 253–326.

[10] D. Westerståhl, Generalized Quantifiers, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Winter 2019 ed., Metaphysics Research Lab, Stanford University, 2019.

[11] R. Montague, The proper treatment of quantification in ordinary english, in: J. Hintikka, J. Moravcsik, P. Suppes (Eds.), Approaches to natural language: proceedings of the 1970 Stanford workshop on Grammar and Semantics, Reidel, Dordrecht, 1973.

[12] D. van Dalen, Logic and Structure, Universitext, fifth ed., Springer-Verlag, 2013.

[13] J. Väänänen, Second-order and Higher-order Logic, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Fall 2021 ed., Metaphysics Research Lab, Stanford University, 2021.

[14] R. Moot, C. Retoré, Classical logic and intuitionistic logic: equivalent formulations in natural deduction, Gödel-Kolmogorov-Glivenko translation, 2016. arXiv:1602.07608.

[15] W. D. Goldfarb, The undecidability of the second-order unification problem, Theor. Comput. Sci. 13 (1981) 225–230. URL: https://doi.org/10.1016/0304-3975(81)90040-2. doi:10.1016/0304-3975(81)90040-2.

[16] L. Henkin, Completeness in the theory of types, J. Symb. Log. 15 (1950) 81–91. URL: https://doi.org/10.2307/2266967.

[17] S. Rudolph, Foundations of description logics, in: A. Polleres, C. d'Amato, M. Arenas, S. Handschuh, P. Kroner, S. Ossowski, P. F. Patel-Schneider (Eds.), Reasoning Web. Semantic Technologies for the Web of Data - 7th International Summer School 2011, Tutorial Lectures, volume 6848 of *LNCS*, Springer, 2011, pp. 76–136. doi:10.1007/978-3-642-23032-5_2.

[18] S. Kripke, Identity and necessity, in: M. Munitz (Ed.), Indentity and Individuation, New-York University Press, 1971, pp. 135–164.

[19] S. Kripke, Naming and necessity, in: D. Davidson, G. Harman (Eds.), Semantics of Natural Language, Reidel, 1972, pp. 253–355.

[20] M. Fitting, R. Mendelsohn, First-Order Modal Logic, Springer Netherlands, 1998.

[21] R. Muskens, A theory of names and true intensionality, in: M. Aloni, V. Kimmelman, F. Roelofsen, G. W. Sassoon, K. Schulz, M. Westera (Eds.), Amsterdam Colloquium 2011, volume 7218 of *LNCS/FoLLI*, Springer-Verlag, 2012, pp. 441–449.

[22] S. Peters, D. Westerståhl, Quantifiers in Language and Logic, Clarendon Press, 2006.

[23] L. Henkin, Some remarks on infinitely long formulas, in: Infinitistic Methods: Proceedings of the Symposium on Foundations of Mathematics, Warsaw, 1959, Panstwowe Wydawnictwo Naukowe / Pergamon Press, Warsaw, 1961, pp. 167–183.

[24] J. Hintikka, G. Sandu, Game-theoretical semantics, in: J. van Benthem, A. ter Meulen (Eds.), Handbook of Logic and Language, North-Holland Elsevier, Amsterdam, 1996, pp. 361–410.

[25] J. Szymanik, Branching Quantifiers, Springer International Publishing, Cham, 2016, pp. 143–162. doi:10.1007/978-3-319-28749-2_9.

[26] S. Hetzl, A. Leitsch, D. Weller, CERES in higher-order logic, Ann. Pure Appl. Log. 162 (2011) 1001–1034. doi:10.1016/J.APAL.2011.06.005.

[27] M. Baaz, A. Lolic, Towards a proof theory for Henkin quantifiers, J. Log. Comput. 31 (2021) 40–66. URL: https://doi.org/10.1093/logcom/exaa071.

[28] J. Avigad, R. Zach, The epsilon calculus, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Center for the Study of Language and Information, 2008. URL: http://plato.stanford.edu/.

[29] S. Chatzikyriakidis, F. Pasquali, C. Retoré, From logical and linguistic generics to Hilbert's tau and epsilon quantifiers, IfCoLog Journal of Logics and their Applications 4 (2017) 231–255.