JOHAN VAN BENTHEM

POLYADIC QUANTIFIERS*

1. Monadic and Polyadic Quantification

Standard generalized quantifiers are of the unary form

$$Qx \cdot \phi(x),$$

with a set-theoretic interpretation of the type '$[[\phi]] \in Q$'. Polyadic quantifiers generalize this to higher arities:

$$Qx_1 \ldots x_n \cdot \phi(x_1, \ldots, x_n).$$

For instance, the following binary form defines the set of all *transitive* binary relations:

$$Qxy \cdot \phi(x, y) := \forall x \forall y(\phi(x, y) \rightarrow \forall z(\phi(y, z) \rightarrow \phi(x, z))).$$

The linguistic uses of the unary notion (introduced in Mostowski (1957)) have been amply demonstrated in the well-known trilogy Barwise and Cooper (1981), Higginbotham and May (1981), Keenan and Stavi (1986). Recently, however, linguists have also turned toward the more general version (due to Lindström (1966)), witness Keenan (1987b) and May (1987). (Compare also Bellert and Zawadowski (1987).) This paper addresses two issues concerning this new development: its empirical motivation, and especially, its theoretical properties.

Genuinely polyadic quantifier patterns are not to be regarded as a mere source of epicycles for the standard semantic accounts of quantification. There is something more at stake, both linguistically and philosophically. What is being revived here is in fact the traditional topic of 'multiple quantification', which was already studied in the Middle Ages by scholastic logicians. Now, the standard historical verdict, put forcefully in Dummett (1973), is that this whole line of research was misguided from the start. And it was Gottlob Frege who finally solved the problem of multiple quantification, precisely by *ignoring* it: one explanation of single quantifiers suffices, when used iteratively in tandem with the

* I would like to thank Sjaak de Mey for some stimulating discussions on these topics, in connection with his forthcoming dissertation on dyadic quantification and related linguistic phenomena.

Linguistics and Philosophy **12**: 437–464, 1989.

syntactic composition of more complex sentence patterns. Briefly, *polyadic quantification is iterated unary quantification*. It is precisely this view which is being challenged by the recent linguistic developments recorded here.

There is also something at stake for the current theory of Generalized Quantifiers in natural language (cf. van Benthem (1986), Part I), which has analyzed such special properties of unary quantifiers as logicality, conservativity or monotonicity. This theory too has a standard Fregean extension to iterated cases, by means of techniques from Categorial Grammar. In particular, one may derive generalized quantifier meanings for transitive sentences

NP1 TV NP2, or Q1A R Q2B:

yielding the well-known wide or narrow scope readings. [The qualification 'standard' refers to the fact that this extension is not ad-hoc, but rather justified as an instance of a very general procedure of type composition: see van Benthem (1986), Chapter 7.] Thus, as we shall see in more detail below, iterated unary quantifier patterns will automatically inherit various denotational properties from their unary components. For instance, *conservativity* for the separate Q1 and Q2 in the above schema will imply, by a simple calculation, conservativity for compound dyadic patterns Q(A, B, R), in a suitable sense to be defined in Section 3. And similarly, monotonicity features of component quantifiers can be related systematically to those of their compounds. By contrast, however, once genuinely polyadic patterns are to be admitted, there will be a need for novel, rather than automatic extensions of existing generalized quantifier theory.

The main contribution of this paper is a modest one, namely to suggest a more systematic logical point of view on the phenomenon of polyadic quantification. Surveying proposed examples of polyadic quantifier patterns in natural language, we find several cases 'around' what might be called the 'Frege Boundary' of iterated unary definability, and a few beyond that. Thus, it becomes of interest to locate that border line more precisely in structural mathematical terms: something which is achieved in the main theorem of Section 4. As a consequence, we can analyze and classify the earlier empirical examples in a more systematic fashion.

In addition, we show how even iterated unary quantifier patterns by themselves give rise to interesting new questions, often having to do with the phenomenon of 'scope'. Drawing upon the existing literature, we characterize various types of scope-free quantifiers, for which iteration is relatively 'loose'. [Another noteworthy topic in this area is the structural definition of the notion of 'case' found in Keenan (1987a), which is based on the above-mentioned mechanism of type change in transitive contexts.]

Finally, we illustrate how the proper perspective on what is going on here should eventually be a more general type-theoretic one – so as to achieve maximal linguistic mileage out of the semantic lessons learnt in this particular instance of polyadicity in natural language.

2. Empirical Evidence for Polyadicity

Leaving aside such technical examples as the initial one in Section 1, what is the evidence for genuine polyadic quantification in natural language? This is one central empirical question to be answered, and we shall review some proposed contenders. One caveat may be necessary here. The central issue is whether or not certain polyadic patterns have a natural decomposition into their unary components – and not so much whether they are *first-order definable.* Indeed, not all Fregean iterations are first-order, nor all genuine polyadics higher-order.

A first, rather obvious example arises from the Fregean procedure itself:

2.1. *Unary Iteration*

Iteration creates complexes such as

$$Q_1 x \cdot Q_2 y \cdot \phi(x, y) \qquad \text{(compare "}\textit{every boy}\text{ loves }\textit{a girl}\text{").}$$

But of course, as was observed already in Section 1, such complexity can be dealt with entirely by a compositional use of the meanings of unary quantifiers.

These iterated cases gain interest, however, with certain additions; such as in the following example brought up by Keenan:

every boy loves a *different* girl.

Here, the meaning is no longer a simple decomposable $\forall\exists$, as the dependency expressed should now be *one-to-one.* Keenan takes the latter to be a genuine binary generalized quantifier.

Still, one might prefer to treat "different" here as a higher-order operator on an ordinary unary iteration – reflecting our intuitive ideas about the compositional structure of this sentence, as being a 'connected' (or 'frozen') iteration.

Further examples of this phenomenon arise through interaction with another ubiquitous linguistic process:

2.2. *Iteration with Anaphoric Links*

Unary iterations can be 'tied together' by anaphoric links. Again, one might prefer to analyze such cases as (higher-order results of transformations applied to) instances of the basic unary pattern. This will work, e.g., with

every boy loves a girl-friend *of his*;

using the unary predicates 'boy' and 'love a girl-friend of oneself'. But it will not work, apparently, with the Bach–Peters type sentences considered in May (1987):

a boy who loved her left the girl who despised *him*.

As May argues, we seem to need quantification over *couples* of individuals here to get the correct reading.

A related perspective is found in Fenstad et al. (1987):

2.3. *Parametrization*

The following 'donkey sentence'

every farmer who owns *a donkey*, beats *it*

can be analyzed as a parametrized unary case 'every AB' with a parameter x:

(every farmer who owns a donkey x)$y \cdot y$ beats(x).

What should this 'parametrization' mean? One idea is to say that every actual value supplied for x turns this into an ordinary unary case; i.e.,

$$\forall x: \forall y((\text{farmer}(y) \ \& \ \text{owns}(y, x) \ \& \ \text{donkey}(x)) \rightarrow \text{beats}(y, x)).$$

But, this fails with a sentence like

most farmers who own *a donkey*, beat *it*:

which does not mean 'for all donkeys: for most farmers . . .'. The better strategy seems to consist in using couples again, and hence polyadic quantifiers:

'every $xy \cdot \ldots$', 'most $xy \cdot \ldots$'.

DIGRESSION. Of course, problems remain on the latter reading too – as the 'most'-sentences now need not imply (the unary reading of)

most farmers who own a donkey, beat a donkey:

which does seem to be a logical consequence of the former sentence, whatever its construal. But, this is not our main concern here. ■

The next example is another case studied by May (1987):

2.4. *Resumption*

The sentence "no one liked no one" has a reading of the form

$$\text{No } xy \cdot \phi(x, y)$$

expressing that no couple (x, y) belonged to $[[\phi]]$. Now, since neither of the two iterated unary readings for the 'no'-quantifiers has this meaning, the binary approach again seems necessary.

Note, however, that there are unary reductions here in a broader sense. Thus,

$$\text{No } xy \cdot \phi(x, y) \Leftrightarrow \text{No } x \cdot \exists y \cdot \phi(x, y)$$
$$\text{One } xy \cdot \phi(x, y) \Leftrightarrow \text{One } x \cdot \exists y \cdot \phi(x, y) \;\&\; \text{One } y \cdot \exists x \cdot \phi(x, y).$$

A similar reduction is possible for "two", "three", etc.; be it with an ever-growing complexity. We shall return to this phenomenon in Section 4.

Finally, we consider the following:

2.5. *Cumulative Readings*

In addition to its two unary decompositions, a sentence like

three girls ate five plums

also has a so-called 'cumulative' reading, in which the total number of plums eaten by (three) girls equals five. The latter reading, noted e.g. in Scha (1984), reduces to neither of the two possible scope readings 'three(five)' or 'five(three)'.

Still, as in the preceding case, there exists a unary reduction in a more general sense, on the following simple pattern:

$$\text{Three girls } x \cdot \exists y \cdot (\text{plum}(y) \;\&\; \text{eat}(x, y)) \;\&$$
$$\text{Five plums } y \cdot \exists x \cdot (\text{girl}(x) \;\&\; \text{eat}(x, y)).$$

Summing up, the claim seems justified that

- there is a good case to be made for the necessity of higher (non-unary) types of generalized quantifier in natural language;
- but, many of these cases are still similar to the standard ones, in that they amount to treating *tuples* of individuals like individuals themselves.
- and also, various more complex unary reductions occur, which deserve special attention.

To get yet higher cases, one should look at genuine *branching* quantification (see Barwise (1979), Sher (1988)), or perhaps at the Keenan type of example and its ilk, which have no Fregean reduction in any obvious sense.

3. Denotational constraints

The preceding discussion at least motivates taking a closer look at the general logical properties of polyadic quantifiers. For convenience, and practical importance, we restrict attention to the *binary case*.

Already on a universe with n individuals, the class of potential binary generalized quantifiers is quite large. Categorially, the type of Q in the schema $Qxy \cdot \phi(x, y)$ is

$$((e, (e, t)), t):$$

and the size of the corresponding denotational domain is $2^{(2^{(n^2)})}$. But, there are some plausible *denotational constraints* here: as was already the raison d'être for the theory of the unary case (see the survey Westerståhl (1986)).

3.1. *Logicality*

The general categorial concept of logicality applies here too (cf. van Benthem (1986), Chapter 3): as *invariance* of Q under permutations of binary relations induced by *permutations of the individuals*. For all such permutations π, one requires that a polyadic quantifier satisfy

$$R \in Q \text{ iff } \pi[R] \in Q, \qquad \text{for all binary relations R.}$$

(Thus, one retains the 'arrow pattern' of the relation, while disregarding the specific individuals occurring at their ends.) To see the effect of this requirement, one must determine the relation

$$R \approx S,$$

defined as '$S = \pi[R]$ for some individual permutation π'. For unary relations R, S, this just amounts to equicardinality. For binary R, S, the behavior of $\approx$ is more complex:

EXAMPLE. With $n = 2$, $\approx$ has 10 equivalence classes. ■

DIGRESSION. There exists a logical characterization of $\approx$:

PROPOSITION. The following are equivalent on a finite universe M:

- $R \approx S$
- $M, R \models \sigma(X)$ iff $M, S \models \sigma(X)$, for all first-order formulas σ in one binary predicate letter X and identity
- the preceding clause only for universal positive first-order σ.

This result may be proved by elementary model theory. But, it still does not produce one single numerical invariant matching $\approx$. ■

To continue, a logical quantifier Q can now be fully specified as the set of $\approx$-equivalence classes accepted by it. Examples of such logical binary quantifiers are

(1) all iterations of logical unary quantifiers,
(2) all resumptive quantifiers reducing to logical unary quantifiers over couples,

but also, e.g., the earlier-mentioned collection of all transitive binary relations.

Behind all these cases lies a general result (see van Benthem (1986), Chapter 7.5):

PROPOSITION. Any predicate over binary relations which can be defined by means of some formula of Type Theory (that is, a full lambda language with identity) using logical parameters only, is itself logical.

A converse holds too. Every logical polyadic quantifier on a fixed finite universe is definable in such a type-theoretical language over that universe. [See van Benthem (1987) for a general connection between logical invariance and type-theoretic definability.]

Finally, it should be observed that the logicality of unary iterations (case 1) is in fact *derivable* from logicality of their unary components in the ordinary sense. Thus, the art of generalization is to see which derivable properties of unary compounds are also plausible for polyadic quantifiers in general.

3.2. *Conservativity*

Eventually, our whole discussion will have to encompass settings where quantifiers may carry *restricting* predicates to subuniverses. The reasons for making this move here are largely analogous to those concerning unary quantifiers [where the general pattern is (QA)B or Q(A, B)]; but there are also some new ones, witness the examples in Section 2.

For instance, resumptives such as

no A likes no B

call for a representation somewhat like this:

$$\text{No} \; \begin{matrix} \text{A} & \text{B} \\ x & y \end{matrix} \cdot \text{L}(x, y).$$

And, the earlier-mentioned donkey sentences are even explicitly of the form

$$\text{All } x^{\text{S}}y \cdot \text{R}(x, y).$$

So, in general, the restriction itself can be a *relation* on the tuple of relevant variables (cf. Higginbotham and May (1981)).

REMARK. Keenan (1987a) shows that the restriction in the first type of example (being technically of type (1, 1, 2)) cannot be naturally reduced to that in the second (which is of type (2, 2)). The obvious move: replacing A, B by the binary relation $A \times B$, has certain pitfalls. ■

There arises a need, then, for a generalization of such 'unary' topics as *conservativity* (see Keenan and Stavi (1986)), and the interplay of restricting and predicative argument positions generally. Van Eyck (1987) presents a first attempt. For instance, in the last-mentioned case of restricted binary forms Q(S, R), Conservativity becomes

$$\text{Q(S, R)} \quad \text{iff} \quad \text{(Q(S, R} \cap \text{S);}$$

and Logicality likewise

$$\text{Q(S, R)} \quad \text{iff} \quad \text{Q}(\pi[\text{S}], \pi[\text{R}]), \qquad \text{for all individual permutations } \pi.$$

Similar definitions are possible for the case with two unary restrictions; where, e.g., Conservativity assumes the form

$$Q(A, B, R) \quad \text{iff} \quad Q(A, B, R \cap (A \times B)).$$

Again, this requirement may be motivated by showing how, at least for the case of unary iterations, it falls out of ordinary conservativity for the unary quantifiers separately:

$$Q^A x \cdot Q^B y \cdot Rxy \text{ iff } Q^A x \cdot Q^B y \cdot (Rxy \; \& \; By) \text{ iff}$$
$$Q^A x \cdot (Q^B y \cdot (Rxy \; \& \; By) \; \& \; Ax) \text{ iff}$$
$$Q^A x \cdot (Q^B y \cdot (Rxy \; \& \; By \; \& \; Ax) \; \& \; Ax) \text{ iff}$$
$$Q^A x \cdot Q^B y \cdot (Rxy \; \& \; By \; \& \; Ax)\text{: i.e.,}$$
$$Q^A x \cdot Q^B y \cdot (R \cap (A \times B))xy.$$

From now on, for reasons of technical convenience, we shall stay with the simpler unrestricted forms in what follows.

4. Locating the Frege Boundary

In the light of the earlier introduction, there is an obvious interest to the exact location of the border-line between unary Fregean iterations and essentially polyadic quantifiers. [Compare also various reductions discussed for linguistic reasons in May (1987), Sher (1988).] Thus, following the earlier approach to logicality, the following question arises:

> Is there also some kind of invariance characterizing the special class of Fregean binary quantifiers which are definable by unary compounds?

And indeed, there is.

4.1. *Necessary Conditions*

We start with a

DEFINITION. A quantifier $Qxy \cdot \phi(x, y)$ is a *unary complex* if it can be defined as a Boolean combination of forms

$$Q_1 x \cdot Q_2 y \cdot \phi(x, y),$$

with Q_1, Q_2 logical unary quantifiers.

First, we isolate an invariance property of such complexes.

DEFINITION. Set $R \sim S$ if, for all individuals x,

$|R_x| = |S_x|$.

Here, R_x stands for $\{y|(x, y) \in R\}$.

A quantifier Q is *right-oriented* if it is closed under the relation $\sim$.

PROPOSITION. All unary complexes are right-oriented.

Proof. For all individuals x, $Q_2 y \cdot Rxy$ holds iff $R_x \in Q_2$, iff (by the definition of $R \sim S$, and permutation invariance for Q_2) $S_x \in Q_2$, i.e., $Q_2 y \cdot Sxy$. But then, $Q_1 x \cdot Q_2 y \cdot Rxy$ if and only if $Q_1 x \cdot Q_2 y \cdot Sxy$. ■

As an application, note that the earlier Transitivity is not unary definable – witness the following counter-example (where $R \sim S$):

(transitive) (non-transitive)

REMARK. This result can be extended to include converse forms of definition $Q_1 x \cdot Q_2 y \cdot Ryx$ – by using an additional requirement concerning predecessors:

$|_x R| = |_x S|$.

E.g., Transitivity will still remain undefinable, as the above R, S also satisfy this additional requirement. ■

4.2. *Sufficient Conditions*

Is the above semantic behaviour also *sufficient* for unary definability? One illustration is provided by the earlier resumptives. These are all right-oriented. [The reason is this. If $|R_x| = |S_x|$ for all x, then $|R| = |S|$.] And in fact, they are all definable by unary complexes:

EXAMPLE. The statement 'Two $xy \cdot Rxy$' is equivalent to the unary complex

$$(P1x \cdot \exists y \cdot Rxy \;\&\; \exists x \cdot P2y \cdot Rxy) \vee$$
$$(P2x \cdot \exists y \cdot Rxy \;\&\; P2x \cdot P1y \cdot Rxy)$$ ■

This observation inspires the following general result.

THEOREM. On any finite universe, a binary quantifier Q is definable by some unary complex if and only if it satisfies the following two conditions:

(i) Q is logical (i.e., permutation-invariant),
(ii) Q is right-oriented.

Proof. 'Only if'. This follows from the preceding observations.
'If'. Suppose that Q satisfies (i) and (ii). Let there be n individuals. The following unary complex defines the quantifier Q:

$$\bigvee_{R \in Q} \bigwedge P n_j x \cdot P j y \cdot R x y;$$

where the conjuncts enumerate all sizes $|R_x|$ occurring in R with their exact multiplicity.

To show that this works, it suffices to check that, if a relation S satisfies this formula, then it must belong to Q. Now, S will satisfy some disjunct, and hence it has the same 'R_x-distribution' as some $R \in Q$. Let π be any permutation of the individuals sending the $n_j x$ having exactly j S-successors to those having exactly j R-successors. Then we have

$$S \approx \pi[S] \sim R.$$

And so, by conditions (i) and (ii), S must be in Q too. ■

In order to illustrate the situation, we consider two previous types of quantification. First, as was observed before, *resumptive* quantifiers are all right-oriented and logical: and indeed, they do possess unary reductions of the above kind.

Next, *cumulative* quantification is not right-oriented, as may be seen in the following picture of girls and plums

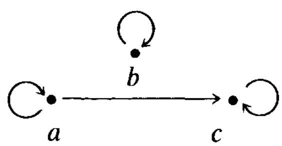
The two relational patterns are ~-connected. And yet, to the left, two girls are eating one plum (cumulatively), whereas, to the right, two girls are eating two plums (again, cumulatively). Nevertheless, it can be shown that cumulative quantification does have the slightly weaker property of *left&right orientation*. That is, it is preserved under the transition from one relation to another having the same number of successors *and predecessors* at each point as the original one. Now, virtually the same proof as above will establish the equivalence of left&right orientation plus logicality with definability in terms of unary complexes admitting both the relation R and its converse. This then explains the unary reducibility of sorts encountered with cumulative quantification in Section 2.

The preceding definability result is only *local*, in some specific universe. But it can probably be extended to provide a characterization of unary definability uniformly in all finite universes.

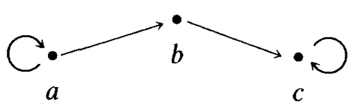
On the other hand, refutations by this method are strong: in that they even refute unary definability within one specific model. Another illustration of this phenomenon is the Keenan quantifier, which we shall now read technically as

> '$\forall x \exists y Rxy$ & R contains a 1–1 function with the same domain'.

The latter statement is true in the following situation (take the identity function):



But it fails in the next situation; although the three corresponding points there have the same numbers of successors and predecessors (i.e., ~ holds):



No one-to-one function can be selected, however, as there would be a clash in the values for *b*, *c*.

Incidentally, the Keenan quantifier is not *first-order* definable in general – and it is not even first-order definable on *finite* universes (as may be proved by a Fraïssé-type game argument).

REMARK. Keenan himself (personal communication) doubts the above higher-order reading for his 'different' sentence. But the pictorial argument presented here also seems to work for a whole range of other meanings for this quantifier combination.

Keenan (1987b) also studies the question of unary reducibility. His notions and results seem somewhat different, however, from those presented here; involving various technicalities. ■

5. Exploring the Realm of Polyadicity

The above methods suggest a more systematic perspective upon polyadic quantifiers, as coming in various natural classes of semantic invariance behaviour. The largest class is that of merely logical ones, being invariant for individual permutations. At the other extreme lies the class of 'resumptive' quantifiers over tuples, which are even invariant for *permutations of pairs of individuals*. Essentially, the latter can only express conditions on the cardinality of the denotation $[[\phi]]$. As every permutation of individuals induces a unique permutation of couples (though not conversely!), this indeed strengthens ordinary logicality.

For a clear non-example of the latter kind, consider the earlier-mentioned Keenan quantifier 'every A R a different B'. It holds in the left-most situation depicted below, but not in its companion (arising from a permutation of couples):

Other, intermediate types of permutation on couples may be used as well to describe important special classes of polyadic quantifiers. Examples can be found in Higginbotham and May (1981), and de Mey (1987) (e.g., in the analysis of *reciprocals*).

EXAMPLE. Here is an illustration from the former paper. Permutations of couples may be induced by individual permutations, as in the definition of Logicality:

$$\pi(a, b) = (\pi(a), \pi(b)).$$

But also, *independent* permutations might be allowed for the two argument positions:

$$\pi(a, b) = (\pi_1(a), \pi_2(b)).$$

Invariance under such *duplex* permutations defines a new class of quantifiers, in between the logical ones and the resumptive cases. Here are some relevant observations:

- $\lambda R \cdot \exists x Rxx$ is logical, but not duplex-invariant.
- $\lambda R \cdot \exists x \forall y Rxy$ is duplex-invariant, but not resumptive.

A more complex example of this kind would be $\lambda R \cdot \forall x \forall y \exists z (Rxz \ \& \ Ryz)$: which is not a 'unary iteration' in the sense of the Section 4. ■

The example also suggests a more formal way of registering the effects of special invariance properties: namely, in terms of their behaviour on standard *first-order* statements about the relation R. As was noted above, *all* such statements are logical. But beyond that, restrictions appeared.

For instance, is there a perspicuous syntactic characterization of those first-order formulas which define duplex-invariant polyadic quantifiers?

DIGRESSION. Sjaak de Mey has suggested that the analysis of Fregean quantification given in Section 4 is itself reminiscent of another type of permutation invariance found with Higginbotham and May. Call a permutation of couples *dependent duplex* if it can be written in the following form, allowing movement of the second argument in dependence on the first:

$\pi(a, b) = (p(a), q_a(b))$,
where p is a permutation of the individuals,
and all q_a are injections defined on R_a.

The exact correspondence is as follows.

PROPOSITION. A binary quantifier is definable by some unary complex if and only if it is invariant for dependent duplex permutations.

Proof. It suffices to show that dependent duplex invariance is equivalent to logicality plus right-orientation. From left to right, logicality is the special case where all q_a equal p. Also, right-orientation follows y letting p be the identity map. From right to left, note that

$$\mathrm{R} \sim \{(a, p^{-1} \circ q_a(b)) \mid (a, b) \in \mathrm{R}\} \quad (=\mathrm{R}^*)$$

and

$$p(\mathrm{R}^*) = \{(p(a), q_a(b)) \mid (a, b) \in R\} = \pi(\mathrm{R}).$$

Then apply right-orientation and locality. ■

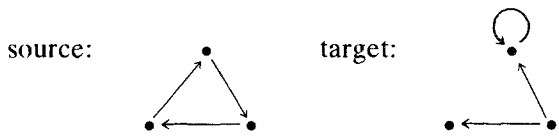
In general, one need not insist on automorphism invariance as the sole means of charting semantic territory. In fact, the earlier notion of *right-orientation* and its obvious dual of 'left-orientation' also define perfectly natural iterative classes, which are both contained in that of the already mentioned 'left&right-oriented' quantifiers.

Going in the opposite direction, however, there is a collapse to one extreme case:

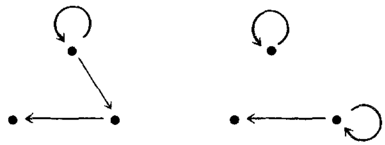
PROPOSITION. Any logical binary quantifier which is both left-oriented and right-oriented is invariant for arbitrary permutations of ordered couples of individuals.

Proof. Here is a sketch. The essential point is that any two binary relations having the same cardinality can be transformed into each other by means of successive steps changing only incoming or only outgoing arrows, which do not disturb numbers of successors or predecessors, respectively, anywhere. [It is instructive to try this with some pictures; e.g., the one used above for the Keenan quantifier.] ■

EXAMPLE. The following strip illustrates the conversion method for relations with equal cardinality.



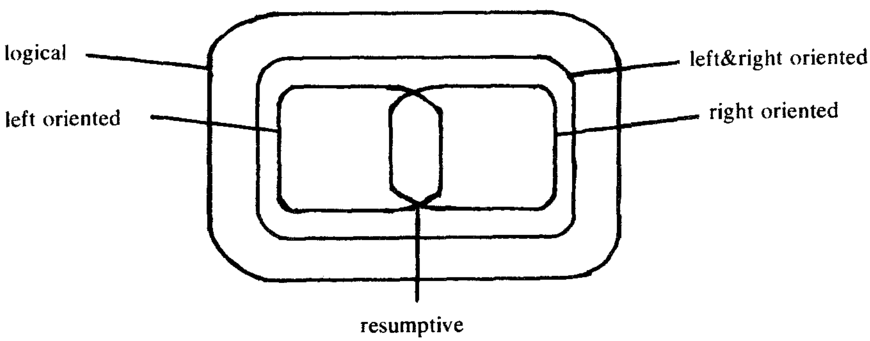
intermediate diagrams:



The first step respects in-degrees, and so does the second, while the final step to the target respects out-degrees ■

Thus, the earlier resumptive quantifiers may be described as being those which are unary iterated themselves, which also having such an iteration for their converse.

The resulting picture for polyadic quantifiers is as follows:



Another approach to charting this territory may be found in Sher (1988). There, the emphasis is placed on progressively more general *schemes of definition* for polyadic quantifiers in a logical representation language. For instance, in the simplest 'independent' case, one has something like the earlier cumulative schema of Section 2:

$$Q1x \cdot \exists y \cdot Rxy \;\&\; Q2y \cdot \exists x \cdot Rxy.$$

Next, one can formulate truly complex cases, such as the following schema inspired by 'positive branching':

$$\exists A \in Q1 \exists B \in Q2 \cdot A \times B \subseteq R,$$

which is then replaced by a 'maximizing' variant for inclusion, and eventually by a very broad schema replacing the double $\exists$ with a more general logical quantification.

In line with the main perspective of this paper, one can analyze these schemata by means of structural semantic conditions. Notably, all proposals satisfy the general constraints of *Logicality* and *Conservativity* (cf. Section 3). On top of these, e.g., the 'independence' schema σ imposes

(1) Invariance under Passive Transformation:

$\sigma(Q1, R, Q2)$ iff $\sigma(Q2, R^{\cup}, Q1)$,
where $R^{\cup}$ is the converse of R.

For instance, observe that, in the cumulative reading, "Three girls ate five plums" is equivalent to its passive form "Five plums were eaten by three girls".

(2) Invariance under Domain/Range Equality:
$\sigma(Q1, R, Q2)$ iff $\sigma(Q1, S, Q2)$,
if $Do(R) = Do(S)$, $Ra(R) = Ra(S)$.

Such special constraints are of independent interest too, witness the discussion of passivization in Section 6.

But also, the more general second schema mentioned above has obvious inferential properties, such as upward *monotonicity* in the three arguments "Q1A", "R" and "Q2B". Thus, on this reading, "At least three boys kissed at least four girls" implies that "At least two boys touched at least two girls".

One interesting observation made by Sher is that surprising *collapses* may occur between the schemata. For instance, 'downward branching' as presented in Barwise (1979), being

$$\exists A \in Q1 \exists B \in Q2 \cdot R \subseteq A \times B,$$

turns out to be equivalent to the iterated 'independent' variant

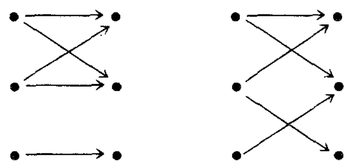
$Q1^{A}x \cdot \exists y \cdot Rxy \;\&\; Q2^{B}y \cdot \exists x \cdot Rxy$
(i.e., '$Do(R) \in Q1A$ & $Ra(R) \in Q2B$').

As an application of the methods in Section 4, we show that, by contrast:

PROPOSITION. The positive branching schema has no definition in the iterated independence format.

Proof. It suffices to exhibit two relations on some domain with the same left and right orientation, only one of which satisfies the positive branching schema. By an earlier theorem, no iterated schema of the 'independent' variety can then define the latter, as such iterated schemes are invariant for this difference.

Let the quantifiers be $Q1 = Q2 =$ *at least two*. Here are the two relevant diagrams:



Note that in- and out-degrees are the same at corresponding points. Yet, only the left-hand diagram satisfies the positive branching schema. ■

Of course, there are many other types of semantic behaviour for polyadic quantifiers which may be studied. For the moment, we hope to have established at least the feasibility of such an investigation.

6. Questions of Iteration

Although unary iterations are not intrinsically polyadic, they do raise some interesting questions of their own, beyond the standard unary framework. Admittedly, it has been emphasized in preceding sections that some of the semantic behaviour of Fregean polyadics is automatically predictable from that of their components. Nevertheless, the phenomenon of iteration also raises several interesting new questions. For instance, several authors have studied *scope* and order of operators in this setting. On the one hand, iteration is itself responsible for the emergence of different scope orderings: but on the other, many expressions involved in this process show a certain freedom of behaviour, which has intrigued quite a few linguists. [In fact, de Mey (1987) takes the *absence* of scope ambiguity to be a reliable test for genuinely polyadic constructions.] Here are some illustrations of this emerging trend.

6.1. *Proper Names*

Zwarts 1986 contains a study of generalized quantifiers that *lack scope with respect to Boolean connectives*. Notably, *proper names* show a collapse of sentence negation and predicate negation:

Mary (doesn't complain) ⇔ Not (Mary complains).

This property is called 'self-duality' in Löbner (1987): $Q = \neg Q \neg$. It seems already so strong that it might completely determine the proper names. But, this is not quite true.

EXAMPLE. Consider a universe $\{1, 2, 3\}$ with a quantifier $Q = \{\{1\}, \{2\}, \{1, 2\}, \{1, 2, 3\}\}$. Then Q is self-dual without even being a filter – and hence it cannot be the denotation of any proper name. ■

But then, proper names also satisfy distribution over conjunctions and disjunctions:

Mary (complains or worries) ⇔
(Mary complains) or (Mary worries).

Using the standard characterization of principal ultrafilters, Zwarts concludes that

PROPOSITION. The proper names are precisely those generalized quantifiers which lack scope with respect to Boolean connectives.

6.2. *Scopeless Quantifiers*

Another notion of scopelessness arises with iterated unary quantifiers in Zimmermann (1987), who considers interchangeability of a quantifier Q with all generalized quantifiers Q′, in the following schema (with either pure or restricted occurrences of Q, Q′):

$$Qx \cdot Q'y \cdot Rxy \quad \text{iff} \quad Q'y \cdot Qx \cdot Rxy.$$

Again, proper names are the prime example here – and Zimmermann proves a converse too:

PROPOSITION. The scopeless quantifiers are precisely the proper names.

To illustrate the kind of reasoning involved, we give a simplified version of his proof. We derive scopelessness with respect to Boolean operations – which reduces the proposition to the preceding result.

Negation. The following semi-syntactic calculation suffices:

$$-X \in Q \text{ iff } Qy \cdot \neg Xy \text{ iff } Qy \cdot \exists z\,(z = z \;\&\; \neg Xy) \text{ iff } Qy \cdot \exists z\neg \cdot (z \neq z \vee Xy) \text{ iff(!) } \exists z\neg \cdot Qy \cdot (z \neq z \vee Xy) \text{ iff } \exists z\neg \cdot X \in Q\ [\text{as } \lambda y \cdot (z \neq z \vee Xy) = \lambda y \cdot Xy] \text{ iff } X \notin Q.$$

Disjunction. Let $\{X_i \mid i \in I\}$ be a family of subsets of the universe. Using the Axiom of Choice, select a subfamily $\{X_j \mid j \in J\}$, together with a set Y of representatives y_j ($j \in J$) such that

(i) the union of the X_i equals that of the X_j, and
(ii) each y_j belongs to a unique X_j (for $j \in J$).

Then, define a binary relation R among individuals as follows:

$$Ryx \text{ if } y = y_j \text{ for some } j \in J \text{ such that } x \in X_j.$$

Note that $x \in \bigcup\{X_i \mid i \in I\}$ iff $x \in \bigcup\{X_j \mid j \in J\}$ iff $\exists y \in Y \cdot Ryx$.

Now calculate as follows:

$$\bigcup\{X_i \mid i \in I\} \in Q \text{ iff } Qx \cdot \exists y \in Y \cdot Ryx \text{ iff(!) } \exists y \in Y \cdot Qx \cdot Ryx$$
$$\text{iff } \exists y_j \in Y \cdot X_j \in Q \text{ (as } \lambda x \cdot Ry_jx \text{ defines } X_j) \text{ iff } \exists j \in J \cdot X_j \in Q \text{ iff } \exists i \in I \cdot X_i \in Q.$$

As to the latter equivalence, one half is obvious, since $J \subseteq I$. Conversely, starting from any given $i \in I$, the selected family $\{X_j \mid j \in J\}$ can be chosen so as to contain X_i. ∎

REMARK. In a sense, proper names are not genuine generalized quantifiers, having been *raised* from type *e* to type ((*e*, *t*), *t*) (cf. van Benthem (1986), Chapter 7). Thus, their freedom of movement in the latter category may be really a sign of 'low status'. It would be of interest to know if something similar holds in general.

Type raising has other uses in this setting too. For instance, in Keenan (1987a), the question is studied which polyadic quantifiers can be viewed as natural lifted versions of monadic counterparts. ∎

Finally, we can push the analysis a bit further by taking up a related question found in van der Does (1988), in the course of a discussion of valid '*exportation* principles' for quantified expressions under perception verbs. In particular, the author notes that all (upward) monotone quantifiers admit of exportation with respect to the existential quantifier:

$$\exists^A x \cdot Q^B y \cdot Rxy \Rightarrow Q^B y \cdot \exists^A x \cdot Rxy,$$

for all monotone generalized quantifiers Q.

Using the earlier method of proof, we may even tighten this to the following

PROPOSITION. The only quantifiers allowing exportation for all monotone quantifiers within their scope are those of the form *some A* for some restriction set A.

Proof. Consider any non-trivial universe. Suppose that a quantifier Q^* allows such exportation: that is, for all binary relations R,

$$Q^*x \cdot Qy \cdot Rxy \Rightarrow Qy \cdot Q^*x \cdot Rxy, \quad \text{for all monotone Q.}$$

[Restricting predicates are omitted here, for convenience.] Then, we can draw the following conclusions:

CLAIM. Q^* is 'splitting': i.e.,

$$\bigcup\{A_i \mid i \in I\} \in Q^* \quad \text{only if} \quad A_i \in Q^* \text{ for some } i \in I.$$

This may be seen from the left-to-right direction of the above Disjunction argument, observing that the quantifier involved in the interchange there is indeed monotone.

CLAIM. Q^* is upward monotone itself.

For, suppose that $A \in Q^*$, A is properly contained in B, but $B \notin Q^*$. Define a monotone quantifier Q and a binary relation R as follows:

$$Q := \{X \mid X \supseteq B\}$$
$$R := (A \times B) \cup ((B - A) \times (B - A)).$$

Then, for $x \in A$, $R_x = B$: which is an element of Q. Next, for $x \notin A$, R_x is either $B - A$ or the empty set $\emptyset$. Now, $\emptyset$ is not in Q, as B is non-empty. Moreover, if $(B - A) \in Q$, then A must be empty (as $(B - A) \supseteq B$). But, this can be ruled out as follows:

If $\emptyset \in Q^*$, then choosing the (upward monotone!) *empty* quantifier Q, and the *empty* relation for R, refutes Exportation.

Therefore, if $x \notin A$, then R_x is not an element of Q. In all, then, we have

$$\{x \mid R_x \in Q\} = A.$$

As A was in Q^*, this says that $Q^*x \cdot Qy \cdot Rxy$. By Exportation, it now follows that

$$Qy \cdot Q^*x \cdot Rxy \qquad \theta$$

Now, for $y \notin \mathrm{B}$, $y\mathrm{R} = \emptyset$: which was outside of Q^*. Next, for $y \in \mathrm{B} - \mathrm{A}$, ${}_y\mathrm{R} = \mathrm{B}$: which was outside of Q^* ex hypothesi. Finally, for $y \in \mathrm{A}$, ${}_y\mathrm{R} = \mathrm{A}$. Together, this implies that

$$\{y \mid {}_y\mathrm{R} \in \mathrm{Q}^*\} = \mathrm{A}.$$

From θ, it then follows that $\mathrm{A} \in \mathrm{Q}$: i.e., $\mathrm{A} \supseteq \mathrm{B}$: *quod non*.

To complete the main proof now, it is easy to check the following identity:

$$\mathrm{Q}^* = \exists^{\mathrm{A}}, \qquad \text{where } \mathrm{A} := \{x \mid \{x\} \in \mathrm{Q}^*\}.$$

[For instance, if $\mathrm{B} \in \mathrm{Q}^*$, then, as $\mathrm{B} = \bigcup\{\{x\} \mid x \in \mathrm{B}\}$, Disjunction implies that some $\{x\}$ is in Q^*, whence $\mathrm{B} \cap \mathrm{A}$ is non-empty.] ■

As a consequence, one may sharpen Zimmermann's result:

COROLLARY. A quantifier admits exportation for arbitrary quantifiers (whether upward monotone or not) if and only if it is a proper name or the empty one.

Proof. 'If'. This follows by direct inspection.

'Only if'. By the preceding result, such a quantifier Q^* must be of the form $\exists^{\mathrm{A}}$ for some A. If A is empty, then Q^* is the empty quantifier. So, suppose otherwise. It suffices to show that A must be a *singleton* set. For the sake of contradiction, then, suppose that $\mathrm{A} = \mathrm{A}_1 \cup \mathrm{A}_2$, with disjoint non-empty parts A_1, A_2. By the definition of Q^*, $\mathrm{A}_1 \in \mathrm{Q}^*$, $\mathrm{A}_2 \in \mathrm{Q}^*$. Now, using the relevant half of the Negation case in the earlier proof, we have that

$$\text{if } \mathrm{A} \in \mathrm{Q}^*, \text{ then } -\mathrm{A} \notin \mathrm{Q}^*.$$

But, if $\mathrm{A}_2 \in \mathrm{Q}^*$, then $-\mathrm{A}_1 \in \mathrm{Q}^*$ (by the monotonicity of Q^*): which is a contradiction. ■

6.3. *Self-Commutation*

Finally, a special important case of scopelessness in the preceding sense is displayed by the *self-commuting* quantifiers of van Benthem (1984):

$$\mathrm{Q}x \cdot \mathrm{Q}y \cdot \mathrm{R}xy \Leftrightarrow \mathrm{Q}y \cdot \mathrm{Q}x \cdot \mathrm{R}xy.$$

Prime examples are the existential and universal quantifiers. For instance, "everyone loves everyone" is equivalent to "everyone is loved by everyone". Thus, in familiar linguistic terms, for these quantifiers, Passivization is a meaning-preserving transformation. [Another interesting linguistic aspect of these quantifiers seems to be that they do not allow genuine *branching* with respect to themselves.] Non-examples are also first-order quantifiers such as *exactly one*, *at least two*. The matter is studied further in Westerståhl (1986a), who proves the following

PROPOSITION. The only (upward) monotone self-commuting quantifiers are *all*, *some*, *true* and *false*.

Proofs of such results have a more combinatorial flavour than those encountered in the standard theory of unary generalized quantifiers. This is already shown in the next

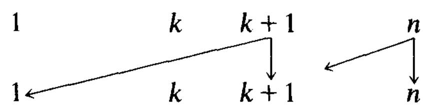
EXAMPLE. On a universe with 2 elements, *exactly one* is still a self-commuting quantifier. ■

By a somewhat laborious calculation, Westerståhl's result can be improved to the following

PROPOSITION. *All*, *some*, *true* and *false* are the only self-commuting *continuous* quantifiers.

Instead of a proof, here is an

EXAMPLE. The downward monotone ('persistent') quantifier Q = *at most k* is not self-commuting. To see this, consider the following picture of a universe with n individuals ($n > k$):



Here, $1, \ldots, k$ have no R-successors; while, for $i > k$, $(i, i), \ldots, (i, i-k) \in R$ (i.e., $k+1$ R-successors each). It is easy to check that

(i) $Qx \cdot Qy \cdot Rxy$, while
(ii) not $Qy \cdot Qx \cdot Rxy$.

[Ad (ii): $1, \ldots, k$ have at most k R-predecessors – e.g., k is preceded by $k+1, \ldots, k+k$ – but so does n, which has only one predecessor.] ■

Finally, self-commuting quantifier pairs may also be redescribed as a special case of *converting* compound binary quantifiers Q, satisfying the condition

$$Q(R) \quad \text{iff} \quad Q(R^{\cup}), \qquad (\text{with } R^{\cup} \text{ the } \textit{converse} \text{ relation of } R).$$

This notion has again interesting connections with earlier ones from Sections 4, 5 and 6. For instance, since $R^{\cup}$ satisfies the same numerical conditions on its set of pairs as R, we have:

> If a binary quantifier is invariant for permutations of pairs, then it is converting.

The converse does not hold in general; witness the case of

$$Qxy \cdot Rxy := \forall x \forall y (Rxy \rightarrow Ryx).$$

But, for self-commuting iterations $Qx \cdot Qy$, the two notions may actually be equivalent.

DIGRESSION. Self-commutation provides one of the many examples where the linguistic study of quantification touches upon issues in other, more mathematical fields of enquiry. Notably, in the foundations of *probability*, there have been studies of so-called 'measure quantifiers', for which self-commutation expresses precisely the central *Fubini Theorem* of probability theory. A concrete example is the complete logic of measure quantifiers discovered by Harvey Friedman (cf. Steinhorn (1985)), whose principles for probabilistic *almost all* are essentially

> upward monotonicity, closure under conjunction,
> containment of all sets $E - \{x\}$ (where E is the total universe, and x any object in E), as well as self-commutation.

Interestingly, continuing the above lines of reasoning, it may be shown that

PROPOSITION. The only generalized quantifier satisfying all Friedman Axioms is the trivial one *true*.

Proof. A derivation will not be given here. See van Lambalgen (1988) for an exposition, as well as further probabilistic background. ■

What this impossibility result shows is, not that there is anything wrong with the above system, but rather that genuine probabilistic quantification cannot employ logicality and full power sets in the previous carefree manner. Non-trivial measure quantifiers will only be invariant for those permutations of the universe E which preserve some suitable *measure* on the latter. Thus, the Friedman Axioms capture something essential about probability. ■

7. A BROADER PERSPECTIVE

7.1. *Categorial Generalization*

Quantifiers form only one special type of expression. Nevertheless, their study often brings to light semantic phenomena of wider significance across natural language. One way of formulating these is in a Categorial Grammar, with an associated Type Theory (see van Benthem (1986), Chapters 3 and 7).

EXAMPLE. Generalized quantifiers have a basic type $((e, t), t)$. But, as we have seen, in transitive contexts, they can also be raised to a type $((e, (e, t)), (e, t))$. Thus, we find 'generalized quantifiers' living among the latter 'relation reducers' too. These will be partly lifted versions of the old ones, recognizable by certain special denotational behaviour - partly also new items. For instance, there is exactly one *logical* relation reducer which behaves precisely like proper names in that it is a Boolean *homomorphism* (cf. Section 6.1): namely the reflexivizer *self* (cf. van Benthem (1988)). ■

But, we can also look for even more general analogies. Notably, does the *iterative versus complex* distinction drawn in the above also make sense with other categorial types of expression? The transitive verb schema studied earlier involves the following types:

(u, v)	(s, x)	(y, z)
NP1	TV	NP2

Here, the functors NP1, NP2 should be able to combine with the TV argument in any order, with the same type of outcome. Therefore, they must have identical types. Moreover, assuming that the final step will be an ordinary application, and the first step a composition ('parametrized application'), there can be only one general pattern which fits:

$$((s, y), y) \qquad (s, (s, y)) \qquad ((s, y), y).$$

Unfortunately, no other contexts of this kind seem to occur in natural language.

But then, we may also consider more general contexts, where operators interact which are quantificational in some more remote sense. For instance, consider the two adverbial modifiers in the complex verb phrase

often	walk	a mile
$((e, t), (e, t))$	(e, t)	$((e, t), (e, t))$

There are two readings here which arise by iteration: 'often (walking a mile)' versus 'often walking' a certain mile. Is there also a truly complex one: say, like the *cumulative* reading of "many hands lifted eleven players", where all those hands *together* lifted the winning team of the Soccer League? The answer appears to be negative. And similar negative, or at least inconclusive outcomes arise in combinations such as

write	five letters	to-day
$(e, (e, t))$	$((e, t), t)$	$((e, t), (e, t))$

Here, the cumulative reading actually co-incides with one of the iterated ones. And, replacing "to-day" by a more quantificational expression will actually force us to make any intended cumulation morphologically visible:

write five letters *in* three hours.

Thus, the question as to linguistic generalizations of our initial situation remains open.

REMARK. An alternative remains, of course, to study the more general role of *compounding particles*, such as "in". Indeed, this would already be relevant for quantifiers themselves – since there too, non-iterative readings often involve such particles:

three boys *together* ate all plums ■

7.2. *Lambda Reduction to Lower Types*

There are also more general *mathematical* questions raised by the earlier account of polyadic quantifiers and their Fregean sub-family. Quite generally,

> Which items in some type of expression are already definable using only items from lower types?

As it stands, this question is still rather vague. But, it can be made more precise using suitable notions of 'definable' and 'lower'. [See van Benthem (1985), Chapter XIX, for one particular general version.] Notably, it makes sense to think of definability in a Lambda Calculus, employing *applications* and *lambdas*.

EXAMPLE. Reducible Noun Phrases. Which items in type $((e, t), t)$ are definable using only items from the lower types (e, t), e and t? Consider any definition for such an item, possibly with parameters. Without loss of generality, the definition can be brought into a *lambda normal form*, leaving no more lambda-conversions to be performed. Moreover, types of variables occurring in the normal form must all be subtypes of $((e, t), t)$. Then, the following facts may be deduced:

> it starts with $\lambda x_{(e,t)}$, followed by an application with types (e, t) and e, or some constant of type t.

Thus, the only genuinely different candidates turn out to be:

$\lambda x_{(e,t)} \cdot x_{(e,t)}(a_e)$ (the 'lifted individual' a_e)
$\lambda x_{(e,t)} \cdot c_t$ ■

In this general perspective, we can also return to the earlier issue of reducible polyadic quantifiers. By a simple calculation, the latter form a dwindling minority in the type $((e, (e, t)), t)$:

$$2^{2^n} \times 2^{2^n} \; (= 2^{2^{n+1}}) \text{ versus } 2^{2^{n^2}}$$

But, what if we allow the two parameters in type $((e, t), t)$ (i.e., the unary quantifiers involved in the reduction) to combine, not just via application, but with full lambda abstraction, as above? Then, in principle, there are *infinitely* many possibilities for schemes of definition. Still, what happens is a collapse to a fixed finite number of combination modes:

PROPOSITION. Let a, b be two items in the type $((e, t), t)$ of some model. The items in the type $((e, (e, t)), t)$ which are lambda/application definable from these reduce to forms $\lambda R \cdot$ followed by a matrix in the following list:

(i) $(\neg)A(\lambda x_e \cdot (\neg)R(x)(x))$
(ii) $(\neg)A(\lambda x_e \cdot (\neg)A((\neg)R(x)))$
(iii) $(\neg)A(\lambda x_e \cdot (\neg)A(\lambda y_e \cdot (\neg)R(y)(x)))$.

Here, 'A' indicates either a or b, and '$(\neg)$' denotes an optional negation.

Proof. What is needed here is some general method of enumeration. We refer to van Benthem (1988), (1989) for a systematic approach, employing context-free grammars describing readings, which are then regularized to obtain a finite state machine producing all relevant lambda forms. A supplementary special-purpose argument is needed to show how the latter reduce to the few cases listed above. ■

Thus, even with full lambda definability, few polyadic quantifiers will be downward reducible to unary ones.

There are obvious generalizations of this kind of analysis to other types. Even though these will not be undertaken here, the present Section may have shown the importance of a more general type-theoretical view of polyadic quantification, and the basic semantic issues raised by it.

References

Bartsch, R., P. van Emde Boas and J. van Benthem, (eds.): to appear, *Meaning and Contextual Expression*, Foris, Dordrecht.

Barwise, J.: 1979, 'On Branching Quantifiers in English', *Journal of Philosophical Logic* **8**, 47–80.

Barwise, J. and R. Cooper: 1981, 'Generalized Quantifiers and Natural Language', *Linguistics and Philosophy* **4**, 159–219.

Bellert, I. and M. Zawadowski: 1987, 'Quantification Feature System as a Quasi-Order of Linguistic Quantifiers', Department of Linguistics (McGill University)/Mathematical Institute (University of Warsaw).

van Benthem, J.: 1984, 'Questions about Quantifiers', *Journal of Symbolic Logic* **49**, 443–66.

van Benthem, J.: 1985, *Modal Logic and Classical Logic*, Bibliopolis/The Humanities Press, Napoli/Atlantic Heights.

van Benthem, J.: 1986, *Essays in Logical Semantics*, Reidel, Dordrecht.

van Benthem, J.: 1987, 'Categorial Grammar and Type Theory', Report 87-07, Institute for Language, Logic and Information, University of Amsterdam.

van Benthem, J.: 1988, 'Logical Constants across Varying Types', Report LP-88-05, Institute for the Study of Language, Logic and Information, University of Amsterdam. [To appear in the *Notre Dame Journal of Formal Logic*.]

van Benthem, J.: 1989, 'The Fine-Structure of Categorial Semantics', to appear in M. Rosner (ed.), *Computational Linguistics and Formal Semantics. Lugano 1988*, Cambridge University Press, Cambridge.

van der Does, J.: 1988, 'A Logic with Generalized Quantifiers for Naked Infinitive Reports', Centre for Cognitive Science, University of Edinburgh.

Dummett, M.: 1973, *Frege. The Philosophy of Language*, Duckworth, London.

van Eyck, J.: 1987, 'Quantifiers', in A. von Stechow and D. Wunderlich (eds.), to appear.

Fenstad, J.-E., P.-K. Halvorsen, T. Langholm and J. van Benthem: 1987, *Situations, Language and Logic*, Reidel, Dordrecht.

Gabbay, D. and F. Guenthner (eds.): 1989, *Handbook of Philosophical Logic*, Vol. IV, Kluwer Academic Publishers, Dordrecht.

Gärdenfors, P. (ed.): 1987, *Generalized Quantifiers: Linguistic and Logical Approaches*, Reidel, Dordrecht.

Groenendijk, J., T. Janssen and M. Stokhof (eds.), 1984, *Truth, Interpretation and Information*, Foris, Dordrecht.

Groenendijk, J., D. de Jongh and M. Stokhof (eds.), 1987a, *Foundations of Pragmatics and Lexical Semantics*, Foris, Dordrecht.
Groenendijk, J., D. de Jongh and M. Stokhof (eds.): 1987b, *Studies in Discourse Representation Theory and the Theory of Generalized Quantifiers*, Foris, Dordrecht.
Groenendijk, J., M. Stokhof and F. Veltman (eds.): 1987*, *Proceedings of the Sixth Amsterdam Colloquium. April 1987*, Institute for Language, Logic and Information, University of Amsterdam.
Harrington, L., M. Morley, A. Scedrov and S. Simpson (eds.): 1985, *Harvey Friedman's Research in the Foundations of Mathematics*, North-Holland, Amsterdam.
Higginbotham, J. and R. May: 1981, 'Questions, Quantifiers and Crossing', *The Linguistic Review* **1**, 41–79.
Keenan, E.: 1987a, 'Semantic Case Theory', in J. Groenendijk et al. (eds.), 1987*, 109–32. [Also in R Bartsch et al. (eds.), to appear.]
Keenan, E.: 1987b, 'Unreducible *n*-ary Quantifiers in Natural Language', in P. Gärdenfors (ed.), pp. 109–50.
Keenan, E. and Y. Stavi: 1986, 'A Semantic Characterization of Natural Language Determiners', *Linguistics and Philosophy* **9**, 253–326. [First version from 1981.]
Lambalgen, M. van: 1988, 'Random Sequences and Generalized Quantifiers', Institute for Language, Logic and Information, University of Amsterdam.
Lindström, P.: 1966, 'First-Order Predicate Logic with Generalized Quantifiers', *Theoria* **32**, 186–95.
Löbner, S.: 1987, 'Quantification as a Major Module of Natural Language Semantics', in J. Groenendijk et al. (eds.), 1987a, pp. 53–85.
May, R.: 1989, 'Interpreting Logical Form', *Linguistics and Philosophy* **12**, this issue, 387–435.
de Mey, J.: 1987, 'Dyadic Quantifiers', Instituut voor Algemene Taalwetenschap, Rijksuniversiteit, Groningen.
Mostowski, A.: 1957, 'On a Generalization of Quantifiers, *Fundamenta Mathematicae* **44**, 12–36.
Scha, R.: 1984, 'Distributive, Collective and Cumulative Quantification', in J. Groenendijk et al. (eds.), 131–58.
Sher, G.: 1988, 'Ways of Branching Quantifiers', Department of Linguistics, Columbia University.
von Stechow A. and D. Wunderlich (eds.): to appear, *Handbook of Semantics*, De Gruyter, Berlin.
Steinhorn, C.: 1985, 'Borel Structures and Measure and Category Logics', in L. Harrington et al. (eds.), pp. 579–96.
Westerståhl, D.: 1986, 'Quantifiers in Formal and Natural Languages', Report 86–55, Center for the Study of Language and Information, Stanford. [In D. Gabbay and F. Guenthner (eds.), 1989, 1–131.
Westerståhl, D.: 1986a, 'On the Order between Quantifiers', in *Logic and Abstraction. Essays dedicated to Per Lindström on his Fiftieth Birthday*, Acta Philosophica Gothoburgiensia I, University of Göteborg, 273–85.
Zimmermann, E.: 1987, 'Transparent Adverbs and Scopeless Quantifiers', in J. Groenendijk et al. (eds.), 1987a, pp. 81–99.
Zwarts, F.: 1986, *Categoriale Grammatica en Algebraische Semantiek*, Dissertation, Nederlands Instituut, Rijksuniversiteit, Groningen.

Universiteit van Amsterdam
Faculteit Wiskunde en Informatica
Roetersstraat 15
1018 WB Amsterdam
The Netherlands