Plural Type Quantification

Proceedings of the Amsterdam Colloquium 1999

Yoad Winter
Technion

This paper introduces some of the main components of a novel type theoretical semantics for quantification with plural noun phrases. This theory, unlike previous ones, sticks to the standard generalized quantifier treatment of singular noun phrases and uses only one lifting operator per semantic category (predicate, quantifier and determiner) for quantification with plurals. Following Bennett (1974), plural individuals are treated as functions of type et. Plural nouns and other plural predicates accordingly denote $(et)t$ functions. Such predicates do not match the standard $(et)((et)t)$ type of determiners. Following Partee and Rooth (1983), type mismatches are resolved using *type shifting operators*. These operators derive collectivity with plurals, keeping the analysis of singular noun phrases, where no type mismatch arises, as in Barwise and Cooper (1981). A single type shifting operator for determiners combines into one reading the *existential* shift and the *counting* (neutral) shift of Scha (1981) and Van der Does (1993). This operator combines the *conservativity* principle of generalized quantifier theory with Szabolcsi's (1997) existential quantification over *witness sets*. The unified lift prevents unmotivated ambiguity as well as the monotonicity ill of existential lifts pointed out by Van Benthem (1986).

Bennett's typing of plurals is based on the distinction between *atomic entities* of type e and *sets* of atomic entities characterized by et functions. This distinction is reflected in the *lexical type* of natural language predicates, which are divided into two subclasses:[1]

1. *Atom predicates* of the lexical type et, including intransitive verbs like *sleep, sing* and *dance* and nouns like *student, teacher* and *committee*.[2]

2. *Set predicates* of the lexical type $(et)t$ (abbreviated ett), including intransitive verbs like *meet, gather* and *disperse* and relational nouns like *friend, brother* and *colleague*.

Non-lexical predicates, inflected with number agreement features, may be of a different type than their lexical type according to the following rules of thumb.

1. Morphologically *singular* predicates are of type et, even when their lexical type is ett.

2. Morphologically *plural* predicates may get the ett type, even when their lexical type is et.[3]

For example, the singular nouns *student* and *friend* and the singular verbs *sleeps* and *meets* are of type et. The plural nouns *students* and *friends* and the plural verbs *sleep* and *meet* are of type ett.

This typing is compositionally derived using the following denotations assumed for the (sometimes covert) number features +SG (singular) and +PL (plural). The +SG feature denotes the function that lowers an ett predicate to type et by unioning the singletons in its extension. The +PL

[1] In Winter (1998,1999) I argue that the distinction between atom predicates and set predicates should not be the classical distributive/collective typology of predicates and suggest a new empirical test for the atom/set classification.

[2] According to the criterion mentioned in footnote 1, and following Barker (1992) and Schwarzschild (1996), "group denoting" nouns like *committee* and *senate* have the same semantic type as other simple nouns.

[3] In fact, as will be remarked below, plural predicates are assumed to be ambiguous between types et and ett.

feature denotes the identity function on ett predicates. Formally:

$$[\![+\text{SG}]\!] \ = \ sg_{(ett)(et)} \ \stackrel{def}{=} \ \lambda \mathcal{P}_{ett}.\lambda x_e.\mathcal{P}(\{x\})$$

$$[\![+\text{PL}]\!] \ = \ id_{(ett)(ett)} \ \stackrel{def}{=} \ \lambda \mathcal{P}_{ett}.\mathcal{P}$$

Note that the $+$SG and $+$PL number features denote functions whose arguments are of type ett. This creates a *type mismatch* with atom predicates, which are lexically of type et. Following Partee and Rooth (1983), this type mismatch triggers a process of *type fitting*. The proposed operator for *predicate fitting* is defined as follows.

$$\mathit{pfit}_{(et)(ett)} \ \stackrel{def}{=} \ \lambda P_{et}.\lambda A_{et}.\emptyset \neq A \subseteq P$$

This is the distributivity operator of Link (1983), which also resolves type mismatches in coordinations like *meet and sleep*, commonly used to argue against Bennett's typing.

The following examples illustrate the use of the number feature denotations and the pfit operator. In (1), the *pfit* operator fits the et type of the atom predicate *student* to the type of the singular/plural number feature. In (2), with the set predicate *meet*, type fitting is unnecessary.

(1) a. $[\![\text{student}]\!] \ = \ [\![\text{student+SG}]\!] \ = \ \mathbf{student}'_{et} \ sg_{(ett)(et)}$ (mismatch)

$sg(\mathit{pfit}(\mathbf{student}')) \ = \ \mathbf{student}'$ (resolution)

b. $[\![\text{students}]\!] \ = \ [\![\text{student+PL}]\!] \ = \ \mathbf{student}'_{et} \ id_{(ett)(ett)}$ (mismatch)

$id(\mathit{pfit}(\mathbf{student}')) \ = \ \lambda A_{et}.\emptyset \neq A \subseteq \mathbf{student}'$ (resolution)

(2) a. $[\![\text{meets}]\!] \ = \ [\![\text{meet+SG}]\!] \ = \ sg_{(ett)(et)}(\mathbf{meet}'_{ett}) \ = \ \lambda x_e.\mathbf{meet}'(\{x\})$

b. $[\![\text{meet}]\!] \ = \ [\![\text{meet+PL}]\!] \ = \ id_{(ett)(ett)}(\mathbf{meet}'_{ett}) \ = \ \mathbf{meet}'$

Note that the composition of the singularity feature denotation sg with the pfit operator gives the identity function on the et domain, as stated below.

Fact 1 *For every* P_{et}*:* $sg(\mathit{pfit}(P)) = P$.

This means that in the proposed mechanism singular morphology has no semantic effect on the lexical denotation of atom predicates, just like plural morphology (whose meaning is id) does not trigger any change in the denotation of set predicates.

Unlike predicates, all lexical *determiners* are assumed to range over atoms: their type is uniformly the standard $(et)(ett)$. In a singular sentence like *no student sleeps* we get the following standard analysis.

(3) $\mathbf{no}'_{(et)(ett)}(\mathbf{student}'_{et})(\mathbf{sleep}'_{et})$
$\Leftrightarrow \ \mathbf{student}' \cap \mathbf{sleep}' = \emptyset$

In the singular sentence *no committee meets*, the denotation of the singular set predicate *meets* is derived using the sg operator, which leads to the following (plausible) interpretation.

(4) $\mathbf{no}'_{(et)(ett)}(\mathbf{committee}'_{et})(sg(\mathbf{meet}'_{ett}))$
$\Leftrightarrow \mathbf{committee}'_{et} \cap \{x_e : \mathbf{meet}'_{ett}(\{x\})\} = \emptyset.$

In plural sentences like *no students slept* or *all the committees met*, we get a type mismatch between the $(et)(ett)$ determiner and the ett plural noun. This is resolved by combining two different processes of *counting* and *existential quantification*. To illustrate the semantic outcome of these processes in full generality, consider the following example.

(5) Exactly five students met.

The exact *counting* of students who met in (5) is achieved in two steps: (i) a *conservativity step*, intersecting the denotation of *meet* with the denotation of *students*; (ii) a *participation step*, unioning both sets of sets that serve as arguments of the determiner. This leads to the following analysis.

(6) $\mathbf{exactly\_5}'(\cup \mathit{pfit}(\mathbf{student}'))(\cup(\mathbf{meet}' \cap \mathit{pfit}(\mathbf{student}')))$
$\Leftrightarrow\ |\{x \in \mathbf{student}' : \exists A \subseteq \mathbf{student}'[x \in A \wedge \mathbf{meet}'(A)]\}| = 5$

In words: the total number of students who participated in student meetings is exactly five.

The counting process is defined in general below as a relation between a determiner and two ett predicates.[4]

Definition 1 *We say that a determiner* $D_{(et)(ett)}$ counts *the* ett *predicates* $\mathcal{A}$ *and* $\mathcal{B}$*, and denote* $\mathbf{count}(D)(\mathcal{A})(\mathcal{B})$*, iff* $D(\cup\mathcal{A})(\cup(\mathcal{A} \cap \mathcal{B}))$ *holds.*

Counting does not yet make sure a meeting of five students actually took place in sentence (5), as intuitively required. For instance, if the only sets of students in the extension of the predicate *meet* are the sets $\{s_1, s_2\}$ and $\{s_3, s_4, s_5\}$, where $s_1, ..., s_5$ are different students, then (6) is formally true whereas sentence (5) is intuitively false, or highly incoherent. To capture this effect, we add an additional existential process to the $\mathbf{count}$ condition. This process is a slightly modified version of the proposal in Szabolcsi (1997) to quantify over *witness sets*. A set W is a *witness* of a determiner D and a set A iff $W \subseteq A$ and $D(A)(W)$ holds. The *witness condition* between a determiner $D_{(et)(ett)}$ and two ett predicates is defined as follows.

Definition 2 *We say that a determiner* D witnesses *the* ett *predicates* $\mathcal{A}$ *and* $\mathcal{B}$*, and denote* $\mathbf{wit}(D)(\mathcal{A})(\mathcal{B})$*, iff either* $\mathcal{A} \cap \mathcal{B} = \emptyset$ *or there exists* $W \in \mathcal{A} \cap \mathcal{B}$ *such that* $D(\cup\mathcal{A})(W)$ *holds.*

In other words: D witnesses $\mathcal{A}_{ett}$ and $\mathcal{B}_{ett}$ iff whenever $\mathcal{A} \cap \mathcal{B}$ is not empty it includes a witness of D and $\cup\mathcal{A}$. For (5), the witness condition derives the following requirement.

(7) $\mathbf{wit}(\mathbf{exactly\_5}'_{(et)(ett)})(\mathit{pfit}(\mathbf{student}'_{et}))(\mathbf{meet}'_{ett})$
$\Leftrightarrow\ [\exists A \subseteq \mathbf{student}'[A \neq \emptyset \wedge \mathbf{meet}'(A)]] \rightarrow \exists A \subseteq \mathbf{student}'[|A| = 5 \wedge \mathbf{meet}'(A)]$

In words: if any student(s) met then there was a meeting of exactly five students.

In Scha (1981) and Van der Does (1993) the existential effect is obtained by a separate reading of determiners, in addition to a "neutral" reading that generates analyses as in (6) (cf. footnote 4). This ambiguity strategy suffers from the fact that neither reading is completely adequate to the semantics of sentences like (5). As mentioned above, the counting reading in (6) ignores the implication in (5) that a meeting of exactly five students took place. The existential reading (7) requires the existence of such a meeting, but creates a more severe problem pointed out in Van Benthem (1986:52-53): "existential" readings like (7) actually allow more than five students to participate in meetings in a sentence like (5). In this way, the existential reading counter-intuitively models all quantifiers as upward monotone.

To solve these problems, the $\mathbf{wit}$ operator is devised so that it combines well with the counting reading into a *unified* operator for quantification with plurals. The conjunction of (6) and (7) correctly analyzes plural sentences like (5) without hurting the (non)monotonicity properties of determiners.

[4]This counting process has semantic implications similar to the "neutral" reading of Scha (1981), which is defined in Van der Does (1993) as a relation $\mathbf{N}$ between a determiner, an $\underline{et}$ predicate and another ett predicate: $\mathbf{N}(D_{(et)(ett)})(A_{et})(\mathcal{B}_{ett}) = \mathbf{count}(D)(\mathit{pfit}(A))(\mathcal{B})$.

This conjunction yields the following general definition of the *determiner fitting operator*, mapping atom-based determiners to set-based determiners.

$dfit_{((et)(ett))((ett)(ettt))} \stackrel{def}{=} \lambda D.\lambda\mathcal{A}.\lambda\mathcal{B}.\mathbf{count}(D)(\mathcal{A})(\mathcal{B}) \wedge \mathbf{wit}(D)(\mathcal{A})(\mathcal{B})$

The $dfit$ operator derives an $ettt$ type quantifier as the denotation of plural noun phrases. This leads to a type mismatch between such quantifiers and ett quantifiers, as for example in the following sentence.

(8) All the students and every teacher smiled.

This kind of type mismatch is resolved by a *quantifier fitting* operator that preserves the distributivity of singular quantifiers and is defined as follows.

$qfit_{(ett)(ettt)} \stackrel{def}{=} \lambda Q_{ett}.\lambda\mathcal{A}_{ett}.Q(sg(\mathcal{A}))$

In words: a quantifier $qfit(Q)$ holds of the sets of sets whose singleton members' union is in Q. Using the $qfit$ operator, sentence (8) is analyzed as follows.

(9) $((dfit(\mathbf{all}'_{(et)(ett)})(pfit(\mathbf{student}'_{et}))) \cap (qfit(\mathbf{every}'_{(et)(ett)}(\mathbf{teacher}'_{et}))))(\mathbf{smile}'_{et})$
$\Leftrightarrow\ \mathbf{student}' \subseteq \mathbf{smile}' \ \wedge\ \mathbf{teacher}' \subseteq \mathbf{smile}'$

Note that *every* and *all* are treated as synonyms, both of them denoting the subset relation between et predicates.

The following facts on the resulting type fitting system show some of its semantic features.

Fact 2 *For all* $D_{(et)(ett)}, A_{et}, B_{et}$*:* $dfit(D)(pfit(A))(pfit(B)) \Leftrightarrow D(A)(A \cap B)$.

This shows that for plural sentences like *all the/no/exactly five students slept*, where both the noun and the verb are lexically of type et, the truth conditions derived by type fitting are equivalent to the standard ones, due to the conservativity of natural language determiners ($D(A)(B) \Leftrightarrow D(A)(A\cap B)$).

Fact 3 *For all* $D_{(et)(ett)} \in \text{MON}{\downarrow}, \mathcal{A}_{ett}, \mathcal{B}_{ett}$*:* $dfit(D)(\mathcal{A})(\mathcal{B}) \Leftrightarrow \mathbf{count}(\mathcal{A})(\mathcal{B})$.

This shows that for sentences with right downward monotone determiners, the existential $\mathbf{wit}$ requirement within *dfit* is redundant. For instance, the meaning derived for sentences like *less than five/no students slept/met* correctly does not require any existence of a set of students that did something. This solves the Van Benthem problem for naive existential techniques.

Fact 4 *For all* $D_{(et)(ett)}, A_{et}, B_{et}$*:* $qfit(D(A))(pfit(B)) \Leftrightarrow D(A)(B)$.

This establishes the preserved distributivity of ett quantifiers *qfit*ted to $ettt$.

Remarks:

1. *Conservativity* is preserved with collective quantification: *D Ns V* is equivalent to *D Ns are Ns that V* also when N is a noun like *friend* or V is a verb like *meet*. The conservativity step within *dfit* captures this fact. Lexical *monotonicity* of determiners is however not always preserved when they combine with ett predicates. For instance, *all the students drank together a whole glass of beer* does not entail *all the rich students drank together a whole glass of beer*, in contrast to the left downward monotone behavior of *all* with et predicates. The $\mathbf{wit}$ part of *dfit* captures this contrast, requiring that in the antecedent there was a set of students (not necessarily rich!) that drank together a whole glass of beer. This correctly cancels the ↓MON entailment.

2. There is a need to allow plural predicates to get also the et type meaning of their singular form. For instance, the sentence *all the committees met* is ambiguous between two readings. One reading requires that there was a joint meeting of all the committees together. Another reading states that every committee had (its own) meeting. The first reading is already captured by the $dfit$ operator, whose application is driven by the ett type of the plural noun *committees*. This type of plural nouns is in turn driven by the denotation of the plurality feature $+$PL. The second reading can be obtained by stipulating that the $+$PL feature is ambiguous between the $id_{(ett)(ett)}$ meaning assumed above and the $sg_{(ett)(et)}$ meaning of the $+$SG feature. Using the latter reading, all plural sentences can also get the standard uniform analysis of quantifiers in Barwise and Cooper (1981), in addition to their lifted meaning as derived in this paper.

References

Barker, C. (1992). Group terms in English: representing groups as atoms. *Journal of Semantics*, 9:69–93.

Barwise, J. and Cooper, R. (1981). Generalized quantifiers and natural language. *Linguistics and Philosophy*, 4:159–219.

Bennett, M. (1974). *Some Extensions of a Montague Fragment of English*. PhD thesis, University of California Los Angeles.

Link, G. (1983). The logical analysis of plurals and mass terms: a lattice theoretical approach. In Bauerle, R., Schwarze, C., and von Stechow, A., editors, *Meaning, Use and Interpretation of Language*. De Gruyter, Berlin.

Partee, B. and Rooth, M. (1983). Generalized conjunction and type ambiguity. In Bauerle, R., Schwarze, C., and von Stechow, A., editors, *Meaning, Use and Interpretation of Language*. De Gruyter, Berlin.

Scha, R. (1981). Distributive, collective and cumulative quantification. In Groenendijk, J., Stokhof, M., and Janssen, T. M. V., editors, *Formal Methods in the Study of Language*. Mathematisch Centrum, Amsterdam.

Schwarzschild, R. (1996). *Pluralities*. Kluwer, Dordrecht.

Szabolcsi, A. (1997). Strategies for scope taking. In Szabolcsi, A., editor, *Ways of Scope Taking*. Kluwer, Dordrecht.

Van Benthem, J. (1986). *Essays in Logical Semantics*. D. Reidel, Dordrecht.

Van der Does, J. (1993). Sums and quantifiers. *Linguistics and Philosophy*, 16:509–550.

Winter, Y. (1998). Atom predicates and set predicates: towards a general theory of plural quantification. In *Proceedings of Semantics and Linguistic Theory, SALT8*.

Winter, Y. (1999). Atom and sets: a characterization of semantic number. Unpublished ms. available at the author's webpage.

Computer Science
Technion
Haifa 32000
Israel
E-mail: winter@cs.technion.ac.il
WWW: http://www.cs.technion.ac.il/~winter