Mathematical Structures in Computer Science (2026), **36**, e16, pp. 1–12
doi:10.1017/S0960129526100577

CAMBRIDGE UNIVERSITY PRESS

PAPER

Mathematical structures in natural language semantics: Zawadowski's contribution to linguistics

Stergios Chatzikyriakidis[1] and Robin Cooper[2]

[1]Philology, University of Crete, Greece and [2]Philosophy, Linguistics and Theory of Science, University of Gothenburg, Sweden

Corresponding author: Robin Cooper; Email: cooper@ling.gu.se

(Received 27 June 2025; revised 19 December 2025; accepted 28 February 2026)

Abstract

We will place Zawadowski's work on linguistic semantics in the historical context of the development of the field. Central to his linguistic work is the combination of model theoretic techniques in set theory with type theoretic techniques combined with a continuation semantics. This leads to elegant general treatments of well-known puzzles in the analysis of meaning in natural languages.

Keywords: Type theory; set theory; dependent types; generalized quantifiers; anaphora; continuations

1. Introduction

We will place Zawadowski's work on linguistic semantics in the historical context of the development of the field. We will describe the combination of set theory and type theory that he uses in Section 2. In Section 3, we will focus on dependent types in type theory and the way they are introduced into set theory. We take up continuation semantics in Section 4 and generalized quantifiers in Section 5. Finally, we will discuss the application of these techniques to the analysis of natural language anaphora in Section 6 and conclude with a general comment (Section 7).

2. Set theory and type theory

The work that founded what is known as *formal semantics* in the field of linguistics is three important papers by the logician Richard Montague: "English as a Formal Language," "Universal Grammar," and "The Proper Treatment of Quantification in Ordinary Language" (all collected in Montague 1974). In particular, the last of these papers, commonly referred to as "PTQ," served as the basis for much of the early work in formal semantics in the 1970s and continues to serve as a basis for most work in mainstream formal semantics in the present day, despite some significant changes and extensions. Montague's central claim was that it is possible to give a semantics for natural languages like English in the same way as logicians give semantics for formal languages like the predicate calculus or higher-order languages such as Montague's own *intensional logic*, which he used in PTQ to characterize the semantics of English. What is meant by semantics here is *model-theoretic semantics*. This involves interpreting expressions of the natural language in terms of objects in set theory (one assumes, Zermelo-Fraenkel with urelements, ZFU, although Montague does not explicitly state this in his linguistic papers). In combination with the set

© The Author(s), 2026. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

theory, he used a variant of Church's (1940) *simple theory of types*. He used two basic types, e (the type of individuals) and t (the type of truthvalues), and closed the set of types under an operation of function type formation. That is, for any types T_1 and T_2, the type $(T_1 \rightarrow T_2)$[1] is the type of total functions from the set of objects of type T_1 to objects of type T_2. In addition, for any type T, he introduced the type $(s \rightarrow T)$[2] to be the type of senses corresponding to T Frege's (1892) *Sinn* which he modeled as functions from pairs of a possible world and a time to objects of type T.

Montague's use of types was mainly to provide a regimen for the syntax of natural languages. The interpretation of the combination of syntactic phrases was based mainly on function application, and the types provided the appropriate function space and guaranteed that the interpretation of well-formed phrases of English involved applying functions to objects in their domain. To take a very simple example, which is in a number of ways a simplification of the linguistic analysis presented in PTQ, consider the verb *barks* in English. We can think of this as denoting a function of type $(e \rightarrow t)$, that is, a function from entities (individuals) to truthvalues. The name *Fido* we could think of as denoting a particular dog, that is, an object of type e. The syntactic combination *Fido barks* can then be interpreted as the result of applying the function denoted by *barks* to the individual denoted by *Fido*. This will either be 1, the truthvalue "True," or 0, "False," that is, an element of type t. The sense of such a sentence is, thus, according to Montague, a function of type $(s \rightarrow t)$, a function from pairs of possible worlds and times to $\{0, 1\}$, that is, the characteristic function of a set of world-time pairs. The functions of type $(s \rightarrow t)$ are what Montague would call *propositions*, referring to the notion of proposition as the sense of a declarative sentence following Frege (1892). There are a number of problems in modeling the intuitive notion of proposition in this way, which have been widely discussed in the philosophical and linguistic literature. (For a recent account with references to earlier literature, see Cooper 2023, Chap. 6.) Not least among these problems is the fact that necessary propositions (those true at all worlds and times), including true propositions in mathematics, are, according to this theory, all identical with each other. Thus, there would be only one true mathematical proposition.

One way out of this dilemma is to take propositions to be *types*, in the sense not of simple type theory but modern or rich type theories deriving from the work of Martin-Löf (1984). The "propositions as types" view comes from the tradition of intuitionistic logic. Its usefulness for the analysis of natural language semantics is discussed by Ranta (1994), Section 2.1.6, and its importance in computer science is discussed by Wadler (2015). In a Martin-Löf type theory, it is natural to have a much larger collection of basic types compared to simple type theory. In addition, there are many ways of constructing types in addition to the function-type formation provided by simple type theory. According to the propositions-as-types principle, a proposition (i.e., a type) is true just in case there is some object of the type and false otherwise. An object of the type can be construed as a proof of the proposition. Martin-Löf calls this a *proof object*, making an important intuitive connection between proof-theoretic and model-theoretic approaches to semantics. Now, we can think of there being many distinct mathematical propositions, all of which are necessarily true in this sense.

This then is attractive as an approach to natural language semantics, but the linguist coming from Montague's classical model-theoretic tradition grounded in set theory has to face the fact that the Martin-Löf approach is intuitionistic and proof-theoretic and indeed has the ambition to replace set theory with type theory as the foundation for mathematics. (See Cooper 2017, for a more detailed discussion.) The question arises whether the aspects of a Martin-Löf type theory that would be attractive for linguistic semantics could be transferred to classical model theory based on set theory. Zawadowski's work addresses this issue. Grudzińska and Zawadowski (2017), for example, present an account of types that includes aspects of both these approaches. On the one hand, the approach is model-theoretic with truth and reference being basic concepts, that is, not

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

following the propositions-as-types idea. On the other hand, types are introduced by having a language with variables and constants over types, and type constructors are allowed that allow more constructed types than the function types introduced in simple type theory. Types are interpreted as sets in the standard set-theoretic sense and not, for example, as intuitive proofs or proof objects in the Martin-Löf sense. Nevertheless, they show that this notion of type admits of a notion of dependent type and context in the sense discussed in Martin-Löf type theories. In Section 3, we discuss the notion of dependent type. We will discuss contexts in Section 6.

3. Dependent types

One of the basic features of systems with the tradition of Martin-Löf is the use of dependent types (Martin-Löf 1975; Martin-Löf 1984). Given a type A and a family of types B(x) indexed by elements x of type A, one can construct dependent types using constructors like Σ and Π, both highly relevant for the study of linguistic semantics. In what follows, we give two very well-known examples of dependent types as these are formulated in that tradition:

Dependent Σ-types. The constructor/operator Σ is a generalization of the Cartesian product of two sets that allows the second set to depend on values of the first. For instance, if $[\![human]\!]$ is a type and $male\colon [\![human]\!] \to Prop$, then the Σ-type $\Sigma h\colon [\![human]\!].\ male(h)$ is intuitively the type of humans who are male.

More formally, if A is a type and B is an A-indexed family of types, then $\Sigma(A, B)$, or sometimes written as $\Sigma x : A.B(x)$, is a type, consisting of pairs (a, b) such that a is of type A and b is of type $B(a)$. When $B(x)$ is a constant type (i.e., always the same type no matter what x is), the Σ-type degenerates into the product type $A \times B$ of nondependent pairs. Σ-types (and product types) are associated with projection operations π_1 and π_2 so that $\pi_1(a, b) = a$ and $\pi_2(a, b) = b$, for every (a, b) of type $\Sigma(A, B)$ or $A \times B$.

The linguistic relevance of Σ-types is very easily understood, given that in its dependent case, Σ-types can be used to interpret linguistic phenomena of central importance, e.g., adjectival modification (Ranta 1994). For example, *handsome man* is interpreted as a Σ-type (1), the type of handsome men (or more precisely, of those men together with proofs that they are handsome):[3]

(1) $\Sigma m\colon [\![man]\!].\ [\![handsome]\!](m)$

where $[\![handsome]\!](m)$ is a family of propositions/types that depends on the man m.

Another very common way that Σ-types have been used is in the study of anaphora, most specifically the proper semantic interpretation of donkey anaphora (Sundholm 1986; Ranta 1994). The idea is that we can make use of the projections of the Σ to get donkey anaphora. Below is the interpretation of the classic donkey sentence using dependent typing:

(2) $\Pi z : (\Sigma x : Man)(\Sigma y : Donkey)(\text{own}(x, y))(\text{beat}(\pi_1(z), \pi_1(\pi_2(z))))$

The idea here is that z is a dependent pair of type $(\Sigma x : Man)(\Sigma y : Donkey)(\text{own}(x, y))$. $\pi_1(z)$ extracts the first component of that dependent pair, that is, the man, while $\pi_1(\pi_2(z))$ is the first component of the second component (the donkey).

This donkey anaphora interpretation has been one of the selling points of using dependent type theories. Extensions of these approaches that take care of both strong and weak readings using strong and weak sums have been proposed by Luo (2021). For a compositional analysis of weak/strong readings within dependently typed semantics, see Tanaka (2021). It is important to

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

note here that Dependent Record Types in the sense of Betarte and Tasistro (1998) are equivalent to Σ-types.[4]

Dependent Π-types. The other basic constructor for dependent types is Π. Π-types can be seen as a generalization of the normal function space where the second type is a family of types that might be dependent on the values of the first. A Π-type degenerates to the function type $A \rightarrow B$ in the nondependent case. In more detail, when A is a type and P is a predicate over A, $\Pi x : A.P(x)$ is the dependent function type that, in the embedded logic, stands for the universally quantified proposition $\forall x : A.P(x)$. For example, the following sentence (3) is interpreted as (4):

(3) Every man walks.

(4) $\Pi x : [\![man]\!].[\![walk]\!](x)$

Π-types are very useful in formulating the typings for a number of linguistic categories like VP adverbs or quantifiers. The idea is that adverbs and quantifiers range over the universe of (the interpretations of) CNs, and as such, we need a way to represent this fact. For this reason, Π-types can be used, universally quantifying over the universe CN. (5) the type for VP adverbs[5] while (6) is the type for quantifiers:

(5) $\Pi A : \mathrm{CN}.\ (A \rightarrow Prop) \rightarrow (A \rightarrow Prop)$

(6) $\Pi A : \mathrm{CN}.\ (A \rightarrow Prop) \rightarrow Prop$

Note that the above types are polymorphic in nature. In general, MTTs support polymorphism, a mechanism that has been argued by researchers like Fox and Lappin (2008) to be needed for NL semantics. Type polymorphism is not available in simple type theories.

Variants of Martin-Löf's TT have been used quite extensively and besides the works already mentioned the interested reader is directed to (references are not exhaustive) Sundholm (1986, 1989); Ranta (1994); Luo (2012); Chatzikyriakidis and Luo (2013); Bekki (2014); Bekki and Mineshima (2017); Chatzikyriakidis and Luo (2020).

3.1 Dependent types in Zawadowski's work

One of the things that is really interesting and innovative in Zawadowski's work is the re-engineering of dependent types in a purely model-theoretic setting. In their work, Grudzińska and Zawadowski (2017), as already mentioned, move away from the classic constructive interpretation of propositions-as-types where proofs inhabit types and enter the set-theoretic, model-theoretic world, where every type is interpreted as a set instead, and dependencies are fibrations between sets.

Contrary to *standard* interpretations of Martin–Löf's Type Theory, where universal quantification is expressed by Π and Σ is a dependent pair, in Grudzińska and Zawadowski (2017) Π types are altogether eliminated. Then, quantificational force and scope are instead encoded by three generalized-quantifier *chain* constructors: the *pack* $(?, \ldots)$, the *sequential* $?\,|?$ and the *parallel* $\square$. Σ is re-purposed as an *escape-dependency* operator that unions whole fibers when anaphoric pronouns must ignore an internal binder.

In the analysis of the discourse "Every man loves a woman. They are smart," the first clause introduces a set M of men who love a woman. For each $a \in M$, there is a fiber $W(a) = \{\, w \in W \mid loves(a, w) \,\}$ of the women loved by a, where W is the set of all women. They then form the dependent sum $\Sigma_{a:M} W(a) = \{\, \langle a, w \rangle \mid a \in M,\ w \in W(a) \,\}$; projecting to the second coordinate yields $\bigcup_{a \in M} W(a)$, the set of *all* women loved by some man, so the plural pronoun *they* quantifies over this entire union. This re-purposes Σ as an *escape-dependency* operator that collapses the

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

internal man-index and updates the discourse context. This is unlike the classical Martin–Löf Σ, which simply constructs dependent pairs without such context-merging semantics.

Context handling. Grudzińska and Zawadowski (2017) depart from standard MLTT context handling by treating contexts as dynamic discourse repositories rather than static typing environments.

Dynamic contexts in Grudzińska and Zawadowski (2017):

$$\text{Initial: } \Gamma_0 = k : \mathit{Kid} \tag{7}$$

$$\text{After "Most kids entered": } \Gamma_1 = k : \mathit{Kid}, t_\phi : T_{\phi,\mathit{Most}_k:K} \tag{8}$$

$$\text{After anaphoric sentence: } \Gamma_2 = \Gamma_1 \cup \{\text{refresh operations}\} \tag{9}$$

where $[\![T_{\phi,\mathit{Most}_k:K}]\!] = \{k' \in [\![\mathit{Kid}]\!] : \mathit{entered}(k')\}$.

In Grudzińska and Zawadowski (2017), contexts are discourse content accumulators, that is, each quantifier pack gives a new dependent type $T_{\Phi,\varphi}$ standing for the semantic contribution of the quantification in case. A refresh operation is capable of adding presupposed types (which may or may not be dependent) or Σ-summaries (typically nondependent unions from dependency-escaping operations) as discourse progresses. As such, depending typing ends up being the primary mechanism for tracking what becomes anaphorically accessible.

Summary As we have seen, the two most used dependent types in MLTT variants are Π and Σ. Π is used for universal quantification, and in linguistic semantics, it is a vehicle for type polymorphism, with uses in quantifiers, adverbs, among other phenomena. Similarly, Σ is used for ordinary dependent pairs and linguistically to express existential quantification and adjectival modification, among others. Donkey anaphora as well as other anaphoric phenomena are handled locally making use of the projections π_1, π_2. In Grudzińska and Zawadowski (2017), Π is eliminated altogether: scope and force are carried by generalized-quantifier *chains* – pack $(?, \ldots)$, sequential $? \mid ?$ and parallel □, while anaphora always ranges over the context-created types $T_{\Phi,\varphi}$. Σ is re-purposed as a "sum over fibres" that flattens dependencies. In "Every man loves a woman. They are smart," individual fibers like $W(\text{John}) = \{\text{Mary, Sue}\}$ get unified via $\Sigma_{a:M} W(a)$ into {Mary, Sue, Lisa}, allowing "they" to quantify over all women loved by any man. More details on the treatment of quantification will be given in Section 5.

In sum, Grudzińska and Zawadowski (2017) depart from using dependent types Π and Σ in the usual way we understand them from MLTT and break from this use by dropping Π completely and turning Σ into a dependency-escape mechanism. Instead, they introduce a new family of dependent types $T_{\Phi,\varphi}$ that serve as discourse content accumulators, with each quantifier pack contributing a specific $T_{\Phi,\varphi}$ type to track its semantic contribution. Dependent typing thus becomes the primary mechanism for tracking anaphoric accessibility, with the repurposed Σ operator creating unions across these $T_{\Phi,\varphi}$ types when anaphoric pronouns need to escape their local dependencies.

4. Continuation semantics

Continuation semantics represents one of the most sophisticated approaches to handling scope phenomena in natural language semantics. Traditional compositional approaches encounter significant difficulties when dealing with quantifier scope ambiguities, especially where surface syntactic order does not correspond to intended semantic scope relationships.

The fundamental insight is that quantified noun phrases do not denote simple semantic objects but rather functions that manipulate the ongoing computation of meaning. This perspective,

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

borrowed from computer science, allows for a principled treatment of scope phenomena without requiring syntactic movement operations.

The continuation monad and scope generation. The basis of continuation semantics is the *continuation monad*:

$$\mathcal{C}A = (A \to t) \to t$$

where a continuation represents the semantic context into which an expression's meaning will be plugged. This allows uniform typing: both *John* and *every student* receive type $(e \to t) \to t$, enabling flexible composition through different combination strategies like pile-up operations.[6]

Over-generation. While continuation semantics elegantly generates multiple scope readings, it faces an overgeneration problem, since it can derive scope readings unavailable in natural language. For example, while *a book by every author* allows inverse scope, *a gift for every child* typically does not permit the same scope reversal.

Various solutions address this issue. Mentioning just a few representative accounts, Barker and Shan (2014); Charlow (2014) introduce *scope islands*, that is, syntactic domains constraining scope-shifting operations, De Groote (2006) uses dynamic state to thread discourse referents, and Bumford and Charlow (2025) explore *algebraic effects* with handlers providing fine-grained control.

Integration with type theory. An important development involves integrating continuation semantics with more sophisticated type systems. Most existing continuation-based approaches operate within simple type theory, relying on the continuation monad's computational machinery to handle phenomena that richer type theories address through their type systems. Bernardy et al. (2021) develop a monadic approach to anaphora resolution using elements of modern type theory, demonstrating alternative ways to integrate type-theoretic machinery with computational effects.

Grudzińska and Zawadowski (2017) take a careful middle ground: rather than abandoning continuation semantics for full-scale dependent type theory, or relying solely on the continuation monad's computational power, they employ continuation semantics with a handle-with-care use of dependent typing. This selective integration allows dependent types to constrain the evaluation context precisely where needed, that is, at lexically triggered extension points, while preserving the continuation framework's elegance. The result is neither purely CPS nor purely type-theoretic, but a principled hybrid that builds semantic/lexical restrictions directly into the type structure of contexts.

Zawadowski's dependent-type continuation passing style. The system in Grudzińska and Zawadowski (2017) employs the classical continuation monad $\mathcal{C}A$ but integrates dependent types into the evaluation context structure. The innovation lies in constraining context growth:

$$\Gamma = x_1 : X_1,\ x_2 : X_2(x_1),\ \ldots,\ x_n : X_n(\vec{x}_{<n})$$

Context Γ grows in exactly two ways:

- ▷ *prepend* constant types (surface scope, always allowed)
- ◁ *append* dependent types $Y(x_{active})$ only for:
  - relational nouns (*representative of*, *mother of*)
  - locative prepositions (*from*, *over*, *in*, *on*)

This lexical constraint ties inverse scope availability to genuine semantic dependencies. When continuation passing style (CPS) combinators operate-wise, quantifiers see only the already-built part of Γ. Without a licensed-dependent extension, inverse scope becomes impossible to type.

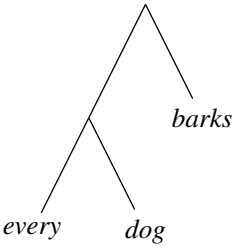
https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

Hence, *a man from every city* (locative *from*) permits inversion, while *a table with every sheet* (instrumental *with*) remains frozen, that is, no QR, LOWER, or covert movement required.

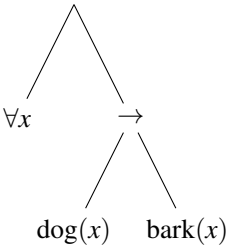
In sum, the account presented in Grudzińska and Zawadowski (2017) achieves good empirical coverage by enriching a standard CPS calculus with a restricted use of dependent typing which is lexically driven.

5. Generalized quantifiers

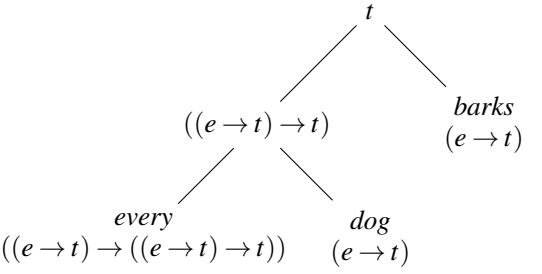
Consider the English sentence *every dog barks*. The constituent structure of this sentence might be represented as:



This contrasts with the structure of the corresponding expression in the predicate calculus:

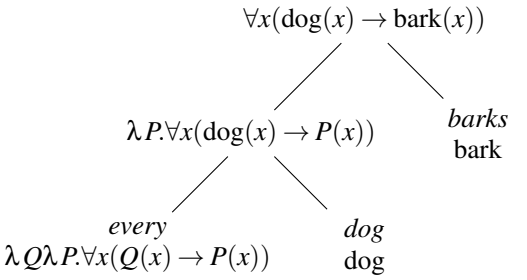


Since Montague wished to give a model-theoretic interpretation defined recursively directly on the structure of the natural language, he needed to show that it was possible to assign an interpretation to the noun-phrase constituent *every dog* and yet show that the complete sentence had an interpretation corresponding to the universally quantified formula of the predicate calculus. His strategy for this was to interpret natural language using the kind of functions provided by a variant of Church's (1940) simply-typed λ-calculus. Here, we simplify Montague's original treatment somewhat for expository purposes. Typing for the constituents is represented by:



https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

The functions assigned to the constituents are those denoted by the λ-expressions below:



Here, predicates are interpreted as the characteristic functions of sets. Thus, the interpretation of the noun-phrase *every dog* corresponds to the characteristic function of a family of sets. For example, if we take Dog_E to be the set of all dogs in a universe of entities E, then the interpretation of *every dog* given above corresponds to the set

$$\{X \subseteq E : \text{Dog}_E \subseteq X\}$$

Such families are used to characterize generalized quantifiers, that is, those which include quantifiers other than the existential and universal quantifiers we know from predicate logic (Mostowski 1957). This idea was first applied to natural language semantics by Barwise and Cooper (1981), who pointed out that natural language quantifiers, $D(A)$, such as every(dog), are *conservative*,[7] that is, they meet the condition:

$$X \in D(A) \text{ iff } X \cap A \in D(A)$$

For discussion of the claim that all natural language quantifiers meet this condition, see Peters and Westerståhl (2006), p. 138f. Using this property, Barwise and Cooper were able to define a notion of *witness set* for quantifiers.

A *witness set* for a conservative quantifier $D(A)$ is any $X \subseteq A$ such that $X \in D(A)$.

Barwise and Cooper saw witness sets as a way of providing a tractable evaluation procedure for natural language sentences containing quantifiers. However, one of the contributions of Grudzińska and Zawadowski (2017) was to show that the quantifiers themselves can be constructed from what Barwise and Cooper call the witness sets, thus obviating the need to distinguish between the sets contained in the quantifier and the witness sets. Thus, the quantifier corresponding to *every dog* would be the singleton set

$$\{\text{Dog}_E\}$$

Given this interpretation of quantifiers, we can check the truth of a quantifier Q combined with a predicate P interpreted as a set X, such as bark in *every dog barks*, by checking whether the set interpreting Q contains a subset of X. Using $[\![\alpha]\!]$ to represent the interpretation of α, we can state this more generally as

$$Q(P) \text{ is true iff } \exists X \in [\![\text{Q}]\!].X \subseteq [\![\text{P}]\!]$$

This will work for conservative quantifiers Q that are *monotone increasing*, that is

$$\text{if } X_1 \sqsubseteq X_2, \text{ then } X_1 \in [\![\text{Q}]\!] \text{ implies } X_2 \in [\![\text{Q}]\!]$$

However, the analysis can be straightforwardly extended to monotone decreasing quantifiers and boolean combinations of monotone quantifiers. See Barwise and Cooper (1981) for discussion and the claim that natural language quantifiers are monotone or boolean combinations of monotone quantifiers. See also Cooper (2023) for a later discussion.

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

Natural language sentences with multiple quantifiers exhibit different ways of combining the quantifiers. For example, the natural reading for *every student wrote an essay* is one where the universal quantifier takes scope over the existential quantifier in the way that is familiar from first-order logic.

$$\forall x\ \text{student}(x) \rightarrow \exists y\ \text{paper}(y) \wedge \text{wrote}(x, y)$$

The order of the quantifier phrases in natural language does not always determine the scope dependencies between quantifiers in the interpretation. For example, *a teaching assistant reads every essay* can be taken to correspond to either of the following two readings:

$$\exists x\ \text{teaching_assistant}(x) \wedge \forall y\ \text{paper}(y) \rightarrow \text{read}(x, y)$$
$$\forall y\ \text{paper}(y) \rightarrow \exists x\ \text{teaching_assistant}(x) \wedge \text{read}(x, y)$$

Grudzińska and Zawadowski (2017) treat these kinds of cases by combining the two quantifiers interpreted as sets of witness sets as described in this section into a single polyadic quantifier (i.e., binding more than one variable) by a process of iteration.[8] The witness set is then not a set of individuals but a set of ordered pairs, a relation between individuals.

There are readings of natural language sentences that are not accounted for by this iterative combination of quantifiers. Consider the sentence *three scientists wrote five papers*. There are readings of this sentence where the quantifiers are combined by iteration (one where there are potentially fifteen distinct papers involved and another where there are potentially fifteen scientists involved). There is, however, also a natural reading (perhaps the one most preferred by speakers of English) where there are just three scientists and five papers, each of the three scientists wrote at least one of the papers, and each of the five papers was written by at least one of the scientists. This is referred to as the cumulative reading that Grudzińska and Zawadowski (2017) obtained by combining the quantifiers with an operation of cumulation.

Grudzińska and Zawadowski (2017) also present a third way of combining quantifiers to obtain branching interpretations. An example of this in English from Barwise (1978) is *Most boys in your class and most girls in my class have all dated each other*. See Peters and Westerståhl (2006) for a useful discussion of the literature on the availability of branching readings in English going back to Hintikka (1973), who claimed that the Henkin quantifier (Henkin 1961) was needed to account for these readings.

It represents an insight that quantifiers can be interpreted directly as sets of what Barwise and Cooper called witness sets, and this facilitates an elegant treatment of the semantics of sentences with several quantifiers combined into polyadic quantifiers and, as we will see in Section 6, the treatment of anaphora, that is, the use of pronouns to refer back to quantifiers.

6. Anaphora

One of the profound contributions of Montague's work on natural language in the late 1960s (collected in Montague 1974) was to show that many cases of pronominal anaphora within sentences can be given a precise model-theoretic treatment in the same manner as bound variables in logic. Thus, Montague's system would allow a successful treatment of the anaphora in *[every dog]$_i$ believes that it$_i$ will catch a cat.*[9]

Important and insightful though this was, it did not extend to a number of other ways in which pronouns in natural language can be related to their quantifier antecedents, in particular, pronouns which are related to previous sentences in a discourse, as in one of the examples discussed by Grudzińska and Zawadowski (2017): *[Most kids]$_i$ entered. They$_i$ looked happy*. This led to a large body of work following Montague on *dynamic semantics,* which introduced the notion of dynamic contexts that are updated by utterances and serve as an environment for the interpretation of pronouns. Notable in this development is much literature on discourse representation

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

theory, for example, Kamp (1981); Kamp and Reyle (1993); Kamp et al. (2011), and Heimian semantics, for example, Heim (1982).

Grudzińska and Zawadowski (2017) combine the insight that natural language quantifier phrases can be interpreted as families of witness sets with a dynamic notion of context adapted from type theories and inspired by the work on a type-theoretic approach to natural language anaphora developed by Ranta (1994). For example, the quantifier phrase *most kids* will be the family of sets whose members are those subsets of ⟦kids⟧, which contain "most" of the members of ⟦kids⟧. What actually constitutes "most" is a matter of some discussion in the literature. It is often assumed to be anything more than half on finite sets, although this seems unintuitive on sets with large cardinality. For example, *most fans in the stadium supported the home team,* which seems unlikely to be an intuitive characterization of a situation where the number of fans supporting the home team was one more than half of those in the stadium. What counts as most seems to depend at least on the size of the sets involved and perhaps also on other aspects of the context. Furthermore, more than half of the characterization does not work immediately for infinite sets: *most natural numbers are not prime*. However, given some way of computing the sets that contain most of the kids, it seems clear that one of those sets needs to be a subset of ⟦entered⟧ in order for the sentence *most kids entered* to be true. The utterance of this sentence will, according to Grudzińska and Zawadowski (2017), add the witnessing set to the context, which will serve as the context for the interpretation of *They looked happy* which will require that each member of this set looked happy. This elegantly treats a common kind of anaphora associated with monotonic increasing quantifiers. This kind of reading is perhaps less readily available with monotone decreasing quantifiers. Consider *Few/less than three students attended the lecture. They were glad that they did*. Another kind of reading, commonly referred to as *complement set anaphora*, is common with monotone decreasing quantifiers, though unavailable with increasing quantifiers: *Few/Less than three students attended the lecture. They were at the beach*. For recent discussion of the analysis of this kind of reading, see Lücking and Ginzburg (2022); Cooper (2023). These treatments are arguably less elegant than the proposal of Grudzińska and Zawadowski (2017), and it would be interesting to explore how their approach could be extended to cover them.

The use of polyadic quantifiers described in Section 5, combined with the use of dynamic context update associated with them, provides an elegant treatment of *quantification subordination* as in *Two dogs chased three cats. They did not catch them*. The trick here is to provide interpretations for the pronouns by extending the contexts with type judgments based on the assembled polyadic quantifier of the previous sentence, exploiting the use of dependent types based on fibers of the relation between dogs and cats introduced. This technique is extended to deal with a range of kinds of anaphora in natural language, including the classic puzzle of *donkey anaphora,* so called because of Geach's (1962) original example, *any man who owns a donkey beats it*. Geach's point was that there is no first-order logic translation of this sentence which associates *any* with a universal quantifier and the indefinite article with an existential quantifier as would normally be expected, and yet has the scoping relation ∀∃, which corresponds to the natural intuitive interpretation of the sentence. Consider an attempt as follows:

$$\forall x\,(\mathrm{man}(x) \wedge \exists y\,\mathrm{donkey}(y) \wedge \mathrm{own}(x, y)) \rightarrow \mathit{beat}(x, y)$$

The problem is that the last occurrence of y remains free. A number of solutions to this puzzle have been proposed in the literature. Classical proposals such as Kamp (1981) and Sundholm (1986) follow Geach in producing an interpretation equivalent to two universal quantifiers:

$$\forall x\,\mathrm{man}(x) \rightarrow \forall y\,\mathrm{donkey}(y) \wedge \mathrm{own}(x, y) \rightarrow \mathrm{beat}(x, y)$$

However, subsequent literature (e.g., Pelletier and Schubert 1989) pointed out the existence of "weak" readings for sentences like *Every person who had a dime put it in the parking meter*, which does not mean that everybody who had a dime put all their dimes in the parking meter, but rather that they put one of their dimes in the meter. On the proposal by Grudzińska and Zawadowski

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

(2017), the difference between the two readings is simply the difference between a universal and an existential quantifier. (See Cooper 2023 for a similar, though arguably less elegant, way of treating the difference between strong and weak readings of donkey anaphora.)

The treatment by Grudzińska and Zawadowski (2017) shows the advantage of a subtle combination of set theoretic modeling techniques with insights from type theoretic approaches in giving a general account of a wide range of anaphoric phenomena.

7. Conclusion

As in logic, set theoretic and type theoretic approaches have been separated in linguistic semantics. Zawadowski's contribution is important in that it shows how elements of type theory can be used to enrich a set theoretic approach. It is not so often that one meets a mathematician with a broad enough view and sufficient insight to see how to combine aspects of them into a single coherent approach. It is downright rare to find such a person with an interest in the analysis of natural language. We are sad to have lost such a friend.

Acknowledgements. Cooper's work was supported by a grant from the Swedish Research Council (VR project 2014-39) for the establishment of the Centre for Linguistic Theory and Studies in Probability (CLASP) at the University of Gothenburg.

Competing interests. The author(s) declare none.

Notes

1 In Montague's notation $\langle T_1, T_2 \rangle$

2 Notated $\langle s, T \rangle$

3 See Chatzikyriakidis and Luo (2017a) for a more elaborate account of adjectival modification using Σ-types.

4 This is not the case for TTR in the sense of Cooper (2023).

5 This was proposed for the first time in Luo (2011).

6 There have been continuation-based approaches proposed for the study of NL semantics. It is not the scope of this paper to dive deep into that literature. However, the interested reader can get a good overview of this quite extensive literature in Bumford and Charlow (2025).

7 Actually, Barwise and Cooper characterized a notion of a quantifier $D(A)$ *living on* the set A, but in subsequent literature, it has become standard to refer to this as conservativity.

8 For discussion of polyadic quantifiers in general, see Peters and Westerståhl (2006).

9 Here, we use the indexing $[\ -\ ]_i$ to indicate the relationship between a pronoun and its antecedent.

References

Barker, C. and Shan, C. (2014). *Continuations and Natural Language*, Oxford University Press.

Barwise, J. (1978). On branching quantifiers in English. *Journal of Philosophical Logic* **8** 47–80.

Barwise, J. and Cooper, R. (1981). Generalized quantifiers and natural language. *Linguistics and Philosophy* **4** (2) 159–219.

Bekki, D. (2014). Representing anaphora with dependent types. In: *LACL 2014*, LNCS, vol. 8535. Springer.

Bekki, D. and Mineshima, K. (2017). Context-passing and underspecification in dependent type semantics. In: Chatzikyriakidis, S. and Luo, Z. (eds.) *Modern Perspectives in Type-Theoretical Semantics*, Springer.

Bernardy, J.-P., Chatzikyriakidis, S. and Maskharashvili, A. (2021). A computational treatment of anaphora and its algorithmic implementation. *Journal of Logic, Language and Information* **30** 1–29.

Betarte, G. and Tasistro, A. (1998). Extension of Martin-Löf 's type theory with record types and subtyping. In: Sambin, G. and Smith, J. (eds.) *Twenty-Five Years of Constructive Type Theory*, Oxford Logic Guides, vol. 36, Oxford University Press: Oxford.

Bumford, D. and Charlow, S. (2025). Effect-driven interpretation: Functors for natural language composition. arXiv preprint arXiv: 2504.00316.

Charlow, S. (2014). On the semantics of exceptional scope. PhD thesis, New York University.

Chatzikyriakidis, S. and Luo, Z. (2013). Adjectives in a modern type-theoretical setting. In: Morrill, G. and Nederhof, J. (eds.) *Proceedings of Formal Grammar 2013*. LNCS, 8036, 159–174. Springer.

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press

Chatzikyriakidis, S. and Luo, Z. (2017a). Adjectival and adverbial modification: The view from modern type theories. *Journal of Logic, Language and Information* **26** (1) 45–88.
Chatzikyriakidis, S. and Luo, Z. (2017b). *Modern Perspectives in Type-Theoretical Semantics*, vol. 98 in Studies in Linguistics and Philosophy. Springer.
Chatzikyriakidis, S. and Luo, Z. (2020). *Formal Semantics in Modern Type Theories*, John Wiley & Sons.
Church, A. (1940). A formulation of the simple theory of types. *Journal of Symbolic Logic* **5** (1) 56–68.
Cooper, R. (2017). *Adapting Type Theory with Records for Natural Language Semantics*. In Chatzikyriakidis and Luo (2017b), 71–94.
Cooper, R. (2023). *From Perception to Communication: A Theory of Types for Action and Meaning*, Oxford University Press. Open access: https://global.oup.com/academic/product/from-perception-to-communication-9780192871312
De Groote, P. (2006). Towards a Montagovian account of dynamics. In: Gibson, M. and Howell, J. (eds.), *Proceedings of Semantics and Linguistic Theory*, vol. 16. Ithaca, NY: Cornell University, 1–16.
Fox, C. and Lappin, S. (2008). *Foundations of Intensional Semantics*, John Wiley & Sons.
Frege, G. (1892). Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik* **100** 25–50. Translated in Geach and Black (1980).
Geach, P. and Black, M. (1980).*Translations from the philosophical writings of Gottlob Frege*, third edition. Oxford, Blackwell.
Geach, P. T. (1962). *Reference and Generality: An Examination of Some Medieval and Modern Theories*, Cornell University Press.
Grudzińska, J. and Zawadowski, M. (2017). *Generalized quantifiers on dependent types: A system for anaphora*. In: Chatzikyriakidis and Luo (2017b), 95–131.
Heim, I. (1982). The Semantics of Definite and Indefinite NPs. PhD thesis, University of Massachusetts at Amherst.
Henkin, L. (1961). Some remarks on infinitely long formulas. In: *Infinitistic Methods: Proceedings of the Symposium on Foundations of Mathematics*, Warsaw, 2-9 September 1959.
Hintikka, J. (1973). Quantifiers vs. quantification theory. *Dialectica* **27** 329–358.
Kamp, H. (1981). A theory of truth and discourse representation. In: Groenendijk, J., Janssen, T. and Stokhof, M. (eds.) *Formal Methods in the Study of Language*, vol. 135 in Mathematical Centre Tracts. Amsterdam, Mathematisch Centrum.
Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*, Dordrecht: Kluwer.
Kamp, H., van Genabith, J. and Reyle, U. (2011). Discourse representation theory. In: Gabbay, D. and Guenthner, F. (eds.) *Handbook of Philosophical Logic*, vol. 15, Springer Science+Business Media B.V.
Lücking, A. and Ginzburg, J. (2022). Referential transparency as the proper treatment for quantification. *Semantics and Pragmatics* **15** (4) 1–58.
Luo, Z. (2011). Contextual analysis of word meanings in type-theoretical semantics. In: *Logical Aspects of Computational Linguistics (LACL'2011)*. LNAI 6736. Springer.
Luo, Z. (2012) Common nouns as types. In: Bechet, D. and Dikovsky, A. (eds.) *Logical Aspects of Computational Linguistics (LACL'2012)*. LNCS 7351. Springer.
Luo, Z. (2021). Donkey anaphora: Type-theoretic semantics with both strong and weak sums. In: Chatzikyriakidis, S. and Osswald, R. (eds.) *Proceedings of the ESSLLI 2021 Workshop on Computing Semantics with Types, Frames and Related Structures*. Utrecht, The Netherlands (online), Association for Computational Linguistics, 45–52.
Martin-Löf, P. (1975). An intuitionistic theory of types: predicative part. In: *Studies in Logic and the Foundations of Mathematics*, vol. 80, Elsevier, 73–118.
Martin-Löf, P. (1984). *Intuitionistic Type Theory*, Naples, Bibliopolis.
Montague, R. (1974). *Formal Philosophy: Selected Papers of Richard Montague*, New Haven, Yale University Press. ed. and with an introduction by Richmond H. Thomason.
Mostowski, A. (1957). On a generalization of quantifiers. *Fundamenta Mathematicae* **44** (1) 12–36.
Pelletier, J. and Schubert, L. (1989). Generically speaking. In: Chierchia, G., Partee, B. H. and Turner, R. (eds.) *Properties, Types and Meanings, Volume II: Semantic Issues*, Dordrecht, Kluwer, 193–268.
Peters, S. and Westerståhl, D. (2006). *Quantifiers in Language and Logics*, Oxford University Press.
Ranta, A. (1994). *Type-Theoretical Grammar*, Oxford University Press.
Sundholm, G. (1986). Proof theory and meaning. In: Gabbay, D. and Guenthner, F. (eds.) *Handbook of Philosophical Logic*, vol. III. Springer.
Sundholm, G. (1989). Constructive generalized quantifiers. *Synthese* **79** (1) 1–12.
Tanaka, R. (2021). Natural Language Quantification and Dependent Types. PhD thesis, Ochanomizu University.
Wadler, P. (2015). Propositions as types. *Communications of the ACM* **58** (12) 75–84.

Cite this article: Chatzikyriakidis S and Cooper R (2026). Mathematical structures in natural language semantics: Zawadowski's contribution to linguistics. *Mathematical Structures in Computer Science* **36**(e16), 1–12. https://doi.org/10.1017/S0960129526100577

https://doi.org/10.1017/S0960129526100577 Published online by Cambridge University Press