Common Nouns as Types

Zhaohui Luo*

zhaohui.luo@hotmail.co.uk

Abstract. When modern type theories are employed for formal semantics, common nouns (CNs) are interpreted as types, not as predicates. Although this brings about some technical advantages, it is worthwhile to ask: *what is special about CNs that merits them to be interpreted as types?* We discuss the observation made by Geach that, unlike other lexical categories, CNs have criteria of identity, a component of meaning that makes it legitimate to compare, count and quantify. This is closely related to the notion of set (type) in constructive mathematics, where a set (type) is not given solely by specifying its objects, but together with an equality between its objects, and explains and justifies to some extent why types are used to interpret CNs in modern type theories. It is shown that, in order to faithfully interpret modified CNs as Σ-types so that the associated criteria of identity can be captured correctly, it is important to assume proof irrelevance in type theory. We shall also briefly discuss a proposal to interpret mass noun phrases as types in a uniform approach to the semantics of CNs.

1 Introduction

It has been proposed that common nouns be interpreted as types, when modern type theories (MTTs) are used to give formal semantics [25]. This is different from the Montague semantics [21], where common nouns are interpreted as predicates. For instance, consider the CN 'book': it is interpreted in the Montague semantics as a predicate of type $e \to t$, while in a modern type theory, it is interpreted as a type. It has been argued that, because CNs are interpreted as types rather than predicates, many linguistic phenomena (eg, copredication), whose formal semantic treatments involve subtyping and have been found difficult in the Montagovian setting, can be dealt with satisfactorily in a straightforward way in MTTs [16]. This has provided some justifications, among others, for MTTs to be employed for formal semantics.

However, one may ask: why can CNs be interpreted as types, but not the other lexical terms such as verbs or adjectives? In the Montagovian setting, CNs, verbs and adjectives are all interpreted as predicates, but in a formal semantics based on MTTs, CNs are interpreted as types and verbs and adjectives are still interpreted as predicates, not as types. The above question may be put in another

* This work is partially supported by the research grant F/07-537/AJ of the Leverhulme Trust in U.K.

D. Béchet and A. Dikovsky (Eds.): LACL 2012, LNCS 7351, pp. 173–185, 2012.

way: *what is special about CNs that merits them to be interpreted as types?* This paper attempts to answer this question and discusses some of the related issues.

We revisit the observation made by Geach and others [11,12,2] that CNs have a special feature that they have their own *criteria of identity*. It is based on this identity criterion that one can decide whether two objects of a CN are the same and it is also based on this that counting, measuring and quantification become possible and meaningful. We argue that it is this special feature that makes it adequate for CNs to be interpreted as types. In constructive mathematics, a set (or a type) is not given solely by specifying its objects, but together with an equality between its objects. In fact, every type in MTTs is associated with such an equality. As illustrated in the paper, when a type is used to interpret a CN, the associated equality corresponds closely to, and formally captures, the criterion of identity of the CN.

In order to correctly capture the criteria of identity for modified CNs in a type-theoretical semantics, where modified CNs are formally represented as Σ-types, one should adopt the principle of *proof irrelevance* (see, for instance, [28] for a recent study). This reflects the basic intuitive idea that, for instance, two handsome men are the same if and only if they are the same man (and it does not matter how one demonstrates that they are handsome). Proof irrelevance identifies the proofs of the same logical proposition and, as a consequence of adopting such a principle, the intended criteria of identity for modified CNs, when formalised as Σ-types, are captured correctly.

Count nouns and mass nouns are both CNs. In formal semantics, one has mostly considered count nouns as primary examples. For mass nouns, although there have been many proposals for their semantic interpretations either as mereological sums or as sets or predicates, no consensus has been reached. In this paper, we take the view that mass noun phrases be interpreted as types and consider a proposal that, when a mass noun is used with a classifier (or measure word), it may be interpreted as a type whose associated equality represents the information given by the classifier. When a mass noun is used without any explicit classifier associated, one may regard it as underspecified in that the criteria of identity can only be determined when more contextual information is available. In MTTs, such underspecification can be represented by means of overloading, as supported by coercive subtyping [17]. As for count nouns, they can be seen as special cases where the measures are obvious and known.[1] This is only a tentative proposal and, when further developed, it may lead to a uniform approach to interpreting CNs, either count or mass, as types.

Introducing the background, §2 contains a brief account of the type-theoretical semantics based on MTTs. In §3, we first introduce the notion of criterion of identity for CNs and then discuss how this is reflected in the formal semantics based on MTTs and how it is linked to the constructive notion of set (or type). Proof irrelevance is discussed in §4, where we show how it can be used in the

[1] This is consistent with the fact that, in the languages with classifiers such as Chinese, a count noun is also used together with a classifier.

	Example	Montague semantics	Semantics in MTTs
CN	man, human	$[\![man]\!],\ [\![human]\!] : e \to t$	$[\![man]\!],\ [\![human]\!] : Type$
IV	talk	$[\![talk]\!] : e \to t$	$[\![talk]\!] : [\![human]\!] \to Prop$
ADJ	handsome	$[\![handsome]\!] : (e \to t) \to (e \to t)$	$[\![handsome]\!] : [\![man]\!] \to Prop$
MCN	handsome man	$[\![handsome]\!]([\![man]\!])$	$\Sigma m : [\![man]\!] .\ [\![handsome]\!](m)$
S	A man talks	$\exists m : e.\ [\![man]\!](m) \& [\![talk]\!](m)$	$\exists m : [\![man]\!] .\ [\![talk]\!](m)$

Fig. 1. Examples in formal semantics

formal semantics based on MTTs to obtain adequate descriptions of modified CNs. Type-theoretical semantics of mass nouns phrases is discussed in §5.

2 CNs as Types in Formal Semantics

In this section, we give a brief introduction to the type-theoretical semantics based on modern type theories (MTTs) [25,16].[2] It is the formal semantics in the style of the Montague semantics [21], but in type theories with dependent types and inductive types, among others, rather than in Church's simple type theory [7] as employed in the Montague semantics. Examples of MTTs include Martin-Löf's predicative type theory [19,20] and the impredicative type theory ECC/UTT [14]. In an *impredicative* type theory like UTT, there is a type $Prop$ of all logical propositions, as to be used in this paper. (This is similar to the simple type theory where there is a type t of truth values.)

In Figure 1, we give some basic examples to illustrate how linguistic categories are interpreted in MTTs and compare them to those in the Montague semantics. A key difference between these two is the interpretation of CNs. In the Montague semantics whose underlying logic can be regarded as 'single-sorted'[3], CNs are interpreted as predicates of type $e \to t$ (and so are verbs and adjectives). In contrast, MTTs can be regarded as 'many-sorted' logical systems where there are many types (eg, inductive types such as the finite types as introduced in Appendix B) that may be used to stand for the domains to be represented; in particular, CNs are interpreted as types [25]. Similarly, modified CNs (MCNs) are interpreted as predicates in the Montague semantics, while they are interpreted by means of Σ-types in MTTs. For instance, in MTTs, 'handsome man' can be interpreted as the type $\Sigma m : [\![man]\!] .\ [\![handsome]\!](m)$. (For Σ-types and their notations, see Appendix A.) Because CNs are interpreted as types, verbs and adjectives are interpreted as predicates over the types (eg, $[\![human]\!]$) that interpret the domains in which they are meaningful: examples are given in Figure 1.

Note that subtyping is crucial for the formal semantics in MTTs. For instance, consider the MTT semantics of the sentence 'A man talks' in Figure 1: for m of type $[\![man]\!]$ and $[\![talk]\!]$ of type $[\![human]\!] \to Prop$, $[\![talk]\!](m)$ is only well-typed because

[2] One may also consult the lecture notes [18], where some informal explanations of MTTs with subtyping are given in the context of formal semantics.

[3] By 'single-sorted' here, we mean that there is only one type e of all entities. Strictly speaking, there is another 'sort'/type t of truth values in Church's simple type theory.

we have that $[\![man]\!]$ is a subtype of $[\![human]\!]$. For MTTs, coercive subtyping as studied in [15] is an adequate framework to be employed for formal semantics [16].

To employ modern type theories, instead of the simple type theory, for formal semantics has many implications, some of which are philosophical, some methodological, and some technical. Technically, for example, the powerful type structures in MTTs give us new useful mechanisms for formal semantics of various linguistic features, examples of which include the use of the dependent Σ-types to interpret modified CNs [25] and the introduction of a type universe CN of the interpretations of common nouns in many linguistic interpretations, including that of adverbs [17].

One of the most notable methodological implications comes from the fact that CNs are interpreted as types, not as predicates. For example, in modelling some linguistic phenomena semantically, one may introduce various subtyping relations by postulating a collection of subtypes (physical objects, informational objects, eventualities, etc.) of the type of entities [1]. It has become clear that, once such subtyping relations are introduced, it is very difficult to deal with some linguistic phenomena such as copredication satisfactorily if CNs are interpreted as predicates as in the traditional Montagovian setting. Instead, if CNs are interpreted as types, as in the type-theoretical semantics based on MTTs, copredication can be given a straightforward and satisfactory treatment [16].

The above methodological advantage may go some way to justify that CNs be interpreted as types (and the employment of MTTs for formal semantics). However, why should CNs be interpreted as types in the first place? Why are they different from the other lexical categories such as verbs and adjectives? After all, in the Montague semantics, CNs, verbs, and adjectives are all interpreted as predicates. Put in another way:

What is special about CNs that merits them to be interpreted as types?

The rest of this paper investigates the related issues and may be regarded as a first step to answer the above question.

3 Criteria of Identity

3.1 An Informal Account

CNs are special, as observed by Geach [11] and others, in that they have the associated *criteria of identity*. Intuitively, a CN determines a concept that does not only have a criterion of application, to be employed to determine whether the concept applies to an object, but a criterion of identity, to be employed to determine whether two objects of the concept are the same. It has been argued that CNs are distinctive in this as other lexical terms like verbs and adjectives do not have such criteria of identity (cf, the arguments in [2]).

The origin of the notion of criteria of identity can be traced back to Frege [10] when he considered abstract mathematical objects such as numbers or lines. For instance, in geometry, the criterion of identity for directions is the parallelism of lines: the direction of line A is equal to that of line B if and only if A and B are parallel. Geach has noticed that such criteria of identity exist for every common noun and is the basis for counting. Gupta [12] has studied this systematically with very interesting examples. For instance, consider the following two sentences:

(1) EasyJet has transported 1 million passengers in 2010.

(2) EasyJet has transported 1 million persons in 2010.

It is easy to see that the first sentence (1) does not imply the second (2), because some people may have traveled more than once by EasyJet in 2010. It has been argued that this is because that the CNs 'passenger' and 'person' have different criteria of identity, which are the basis for counting and have led to such phenomena [11,12,2].[4]

It may be worth noting that the notion of the criterion of identity is context sensitive. In other words, what a CN means depends on the context in which it is used. For instance, consider the word 'student'. In the following sentences, the associated criteria of identity can be different, because in (3) John may have taught several classes and it may be reasonable to count a student in two different classes twice, while this is not the case in (4) where one would say that in that case 'student' seems to have the same criterion of identity with 'person'.

(3) John taught 500 students last year.

(4) 1000 students have applied for campus cards last year.

A close link between the notion of criterion of identity and the constructive notion of set (type) can be established. In constructive mathematics, a set is a 'preset', which gives its application criterion, together with an equality, which gives its criterion of identity that determines whether two objects of the set are the same [5,4]. Modern type theories such as Martin-Löf's type theory [19,20] were originally developed for formalisation of constructive mathematics, where each type is associated with such an equality or criterion of identity. In the following, we shall first consider the formalisations in MTTs of the above example of passengers to demonstrate how the criteria of identity are reflected in such formal representations, and then discuss the link to constructive mathematics in more details.

3.2 Formalisation of an Example

The above example about 'passengers' can be formalised in the MTTs. Let T be the type of the journeys in concern (eg, the journeys that one may make via EasyJet in 2010). We shall consider two different formal presentations of $Passenger[T]$, the type of passengers in journeys of type T, one using finite types and the other considering proof irrelevance.

[4] There have been arguments against the idea of criteria of identity. For instance, Gupta [12] mentioned that one might consider some ontological arguments and Barker [3] has argued against it on the grounds that the linguistic phenomena could better be explained by means of pragmatics. The author believes that the notion of criteria of identity still offers the best explanations.

Representation using finite types. The type $Passenger[T]$ can be defined as the following Σ-type:[5]

$$Passenger[T] = \Sigma p : Person.\ Journey[T](p),$$

where $Person$ is the type of persons and $Journey[T](p)$ the finite type of journeys in T that the person p has made. (See Appendix B for the formal definition of finite types.) In other words, a passenger is a person together with a journey he/she made and, formally, this is represented as a pair (p,t) of type $Passenger[T]$, where t is a journey that the person p has made. There are two points to note about this definition:

1. If $Journey[T](p)$ is empty (the finite type with no object), p has not made any journey in T and is hence not a passenger by definition (there is no passenger (p,t) of type $Passenger[T]$ because there is no such a t of type $Journey[T](p)$.)
2. Formally, (p,t) and (p',t') are equal passengers if, and only if, $p = p'$ and $t = t'$.[6] As passengers, 'John at journey t' and 'John at journey t'' are only equal if t and t' are the same journey.

This last note about equality between passengers is important. It is different from that between persons: the same person making different journeys is regarded as different passengers.

Representation assuming proof irrelevance. Another representation assumes proof irrelevance:

$$\frac{\Gamma \vdash P : Prop \quad \Gamma \vdash p : P \quad \Gamma \vdash q : P}{\Gamma \vdash p = q : P}$$

which intuitively says that every two proofs of the same logical proposition are equal. (See §4 for further discussions on proof irrelevance.) With proof irrelevance, the following Σ-type can be used to represent the type of passengers:

$$Passenger[T] = \Sigma p : Person. \Sigma t : T.\ J(p,t),$$

where $J : Person \to T \to Prop$ is the predicate such that the predicate $J(p)$ of type $T \to Prop$ represents the set of journeys that p has made. It is straightforward to show that two passengers are the same if and only if they are the same person on the same journey. In symbols, $(p,t,v) = (p',t',v')$ if, and only if, $p = p'$ and $t = t'$, because by proof irrelevance, we always have $v = v'$ when the other two components are the same.

[5] An alternative notation for the Σ-type $Passenger[T]$ is $\Sigma(Person, Journey[T])$. See Appendix A for a brief introduction of Σ-types.

[6] Note that, for any $p : Person$, $Journey[T](p)$ is a type, not a logical proposition. Therefore, this is the case even when we have proof irrelevance, as to be discussed in §4.

Remark 1. Both of the above representations give the correct criterion of identity. Intuitively, they give rise to the same criterion of identity between passengers as intended. Other formalisations are also possible. However, it is easy to arrive at some unintended formulations. For instance, it might be tempting to say that 'a passenger is a person who has made one or more journeys'. This would lead to the formalisation of the type of passengers as a Σ-type

$$\Sigma p : Person.\ Travelled[T](p),$$

where $Travelled[T](p)$ is a logical proposition expressing that p has made some journeys in T. Such a formulation does not capture the intended criterion of identity between passengers, no matter whether we assume proof irrelevance or not. For instance, if we do, we have that two objects of the above type ('passengers') are the same if and only if they are the same person who has travelled, because the proofs that the person has travelled are identified. Therefore, such a formal representation would not have captured the intended criterion of identity correctly. □

3.3 Constructive Notion of Set or Type: Further Remarks

In constructive mathematics, the notion of set is associated with an equality (an equivalence relation) [5]. As Beeson [4] puts it, it is a 'preset' together with an equality. A preset X is given by its criterion of application that determines whether an object is in X, while the associated equality determines whether two objects of the set are the same. Unlike classical mathematics, there is no universal equality that can be applied to all objects; instead, different sets are associated with different equalities. When CNs are interpreted as sets, this is directly linked to the notion of criteria of identity in that different CNs may have different criteria of identity.

In modern type theories, a type is a constructive set.[7] For instance, in Martin-Löf's type theory, a type is specified by making clear the following simultaneously:

1. What are the canonical objects of the type?
2. Under what conditions are two canonical objects equal?

Please note that, for completely presented types, these two are enough to determine both criterion of application and criterion of identity.

In MTTs in general, every type is associated with its own equality. For instance, the equality for $\Sigma x{:}A.\ B(x)$ is: for any of its two objects (a, b) and (a', b'), they are equal only when $a = a'$ in type A and $b = b'$ in type $B(a)$ (please note that B is dependent on the objects of A; in this case, it is a). According to

[7] We are rather imprecise here. Types in MTTs are also *restricted* sets with further properties. For example, a type can be inductively defined and, if so, it has many special properties that are not shared by all types. Also, in Martin-Löf's type theory, every type is *completely presented* in the sense that, informally, its criterion of application can be evidenced by computation. The author thinks that such a requirement for objects to be completely presented should not be imposed on linguistic interpretations when MTTs are used for formal semantics.

this definition of equality for Σ-types, both of the above representations for $Passenger[T]$ correctly capture the criterion of identity between passengers as intended. Please note that, in the second representation above, we have to assume proof irrelevance for, otherwise, the representation would not capture the criterion of identity between passengers correctly: if we do not have proof irrelevance, the 'passengers' (p,t,v) and (p,t,v') can be different because the proofs v and v' that p made the journey t are different, although they should be the same passenger.

Remark 2. It seems that the close link between CNs with criteria of identity and types with associated equalities is one of the instances where principles in constructive mathematics can be successfully applied to linguistic semantics in an interesting way. This is reflected above when constructive types are used to represent the semantics of CNs. Further studies of the use of MTTs for formal semantics may shed more light in this respect. □

4 Proof Irrelevance and Identity for Modified CNs

For modified CNs, it is often the case that the criteria of identity are inherited from those before modification. For example, two 'handsome men' are the same if, and only if, they are the same man. In such situations, the adjective used to modify the CN has no effect on the resulting criterion of identity for the modified CN. This should be captured faithfully in a semantic framework.

In MTTs, it has been proposed that modified CNs be interpreted as Σ-types [25].[8] For instance, for $[\![man]\!] : Type$ and $[\![handsome]\!] : [\![man]\!] \to Prop$, the interpretation of 'handsome man' is the following Σ-type:

$$[\![handsome\ man]\!] = \Sigma m : [\![man]\!] .\ [\![handsome]\!](m).$$

However, in type theories (as those implemented in the proof assistants such as Agda, Coq and Lego), the notion of equality between objects of such Σ-types does not capture the intended criteria of identity. In the above example, for the representations (m,h) and (m',h') of two handsome men to be equal, we require not only that the two men m and m' be equal, but that the proofs h and h' be equal as well. But this is usually not the case (there can be more than one way to demonstrate that a man is handsome)!

In order for such Σ-type representations to be faithful in capturing the intended criteria of identity, we should employ *proof irrelevance*, which intuitively says that every two proofs of the same logical proposition are equal. In an impredicative type theory such as UTT, the formal rule for proof irrelevance is (as repeated from §3.2):

$$\frac{\Gamma \vdash P : Prop \quad \Gamma \vdash p : P \quad \Gamma \vdash q : P}{\Gamma \vdash p = q : P}$$

[8] Gupta [12] has suggested a special form of formulae $(K,x)A$, called *restrictions*, for modified CNs. Linking formulae to types, we can easily see the close correspondence between $(K,x)A$ and $\Sigma x : K.\ A$.

Proof irrelevance has been studied for impredicative type theories. For instance, a set-theoretic proof irrelevant model was given in [9] and one can find a recent study on this in [28], where the author has employed the untyped notion of conversion instead of a judgemental equality.

In the above example about handsome men, if $m = m'$, the two proofs h and h' prove the same logical proposition $[\![handsome]\!](m)$ and are hence equal. As a consequence, under this representation, two handsome men are the same if, and only if, they are the same man. In other words, under the assumption of proof irrelevance, the proposed representations of modified CNs by Σ-types do capture the intended criteria of identity.

It is also worth noting that, although proof irrelevance can be considered for impredicative type theories directly as above, it is unclear how this can be done for predicative type theories. For instance, in Martin-Löf's type theory [19,20], propositions are identified with types. Because of such an identification, one cannot use the above rule to identify proofs, for it would identify the objects of a type as well. Put in another way, proof irrelevance is incompatible with the identification of propositions and types. In order to introduce proof irrelevance, one has to distinguish logical propositions and types (see, for example, [14]).

Remark 3. Proof irrelevance is very interesting and has several important applications. Besides the above application in formal semantics, it is also employed in several other fields, including dependently-typed programming (see, for example, [28] for some relevant discussions). □

5 Semantics for Mass Nouns with Classifiers

Common nouns include both count and mass nouns. It seems clear how to consider the criteria of identity for count nouns, but less clear for mass nouns. As for the semantics of mass nouns, scholars have rather different opinions and there seems to be no consensus on this matter.[9] The proposed semantic theories on mass terms fall into two camps: those based on the idea that mass nouns denote *mereological sums*, as advocated by Quine [24] and others, and those based on ideas that they denote *sets*, as considered by people like Laycock [13] (see [6,29] for some rather comprehensive analysis of early work on this).

Here, based on the idea that CNs denote sets/types with associated criteria of identity, I offer a tentative proposal to suggest how some, if not all, of the uses of mass noun phrases may be interpreted. Mass nouns are often used together with measure words or classifiers, as in the following sentences about the mass noun 'water':

(5) John drinks a glass of water.

(6) He has fetched two buckets of water from the river.

[9] One may quote several references, among many, including Quine's remarks on 'divided reference' [24], Strawson's on 'individuals' [26], Baker's on measures [2], Bunt's work on ensemble theory [6] and Nicolas' work on plural logic [22].

The measure words such as 'glass' and 'bucket' provide information for the criteria of identity in these mass noun phrases. It has been proposed [26,8] that mass terms should be understood as elliptical for some phrases with classifiers. For instance, 'water' would be elliptical for 'drop of water', 'glass of water', 'bucket of water', 'body of water', etc. These latter phrases with classifiers are semantically easier to be interpreted as sets and, in particular, it is clearer what the criteria of identity should be. In a type-theoretical semantics based on MTTs, they can be interpreted as types. For example, the mass noun phrase 'glass of water' can be interpreted as a type and the above sentence (5) can be given the following semantics:

(7) $\exists w : [\![glass\ of\ water]\!]\ .\ [\![drink]\!](j, w)$

In such cases, the criteria of identity are provided by the measure words (see [2] for relevant remarks).

In languages like English, count nouns are usually[10] used without measure words (we say 'a table' rather than 'a CLASSIFIER table'). A plausible explanation for this is that the criterion of identity for a count noun, when it is used without a measure word, is 'obvious' or already built in the noun itself. This suggests a uniform way to approach the semantic interpretations of count and mass nouns: they are all interpreted as types and, in the case of mass noun phrases, the measure words in the phrase provide us the information for the criteria of identity while, in the case of count nouns, the counting principle is usually obviously given by the noun itself. Put in another way, this uniform approach is based on the idea that the 'obvious' measures for count nouns are just special cases of the numerals-measures-CN pattern for mass noun phrases such as 'a glass of water' in (5) and 'two buckets of water' in (6).

Such an approach is also informed by considering the languages with classifiers such as Chinese, where every noun is usually used together with a classifier or measure word, even for the count nouns in an English-like language. In Chinese, instead of saying 'a book', one says 'a CLASSIFIER book', where CLASSIFIER is a suitable measure word for 'book'. Some people take the view that the Chinese language does not have count nouns – every CN in Chinese is a mass noun whose uses are often coupled with classifiers. This is consistent with the uniform approach that both count and mass noun phrases are interpreted as types.

When a mass noun is not explicitly used together with a measure word, it becomes a context-dependent matter to determine which measure should be used. There have been some criticisms on the proposal that mass nouns be interpreted as sets (see, for example, [23]). One of the main criticisms is that, when a mass noun is used without a measure word explicitly given, such context dependency seems to amount to a difficulty in considering a 'general translation procedure' to give the semantics for any mass noun [23]. To deal with such potential difficulties, we consider one proposal: in many situations where a mass noun is used without a measure word, it can be seen as *underspecified* and, according

[10] Occasionally, measure words are also used for count nouns in, for example, 'a deck of cards'.

to further enrichments of the contexts, its meaning can then be determined. In MTTs, such underspecified phrases may be given meanings by means of *overloading*, supported by coercive subtyping [17].

The proposal to interpret both count and mass CNs as types requires further studies, both for its justification in general and for the possibility of using the idea to implement reasoning systems that involve mass nouns on the computer.

6 Conclusion

This paper has investigated the idea that CNs have the meaning component of criteria of identity and can be semantically represented by means of types in MTTs. We have pointed out that, in order to interpret CNs adequately, especially to interpret modified CNs by means of Σ-types, the principle of proof irrelevance should be adopted and, furthermore, this principle can only be properly formulated in a type theory where there is a clear distinction between logical propositions and data types (eg, as in most of the impredicative type theories or logic-enriched type theories), but not in those type theories such as Martin-Löf's type theory where there is no such a distinction.

One of the interesting issues is to study generalised quantifiers based on MTTs (see [27] for an early investgation of this in Martin-Löf's type theory.) In this setting, the generalised quantifiers such as 'every' and 'most' have the following type:

$$\Pi A : \textsc{cn}.\ (A \rightarrow Prop) \rightarrow Prop,$$

where CN is the type universe of the interpretations of common nouns. In other words, a generalised quantifier takes as its arguments a type that interprets a CN and a predicate over the type. This is rather different from the 'symmetric' notion of GQs in the Montagovian setting where GQs are concerned with two predicates. This is obviously an interesting topic to study about the formal semantics based on MTTs and requires future investigations.

Acknowledgements. Thanks to Robin Adams, Stergios Chatzikyriakidis and members of the 'Type Theory and Applications' research group at RHUL for helpful discussions.

References

1. Asher, N.: Lexical Meaning in Context: A Web of Words. Cambridge University Press (2011)
2. Baker, M.: Lexical Categories: Verbs, Nouns and Adjectives. Cambridge University Press (2003)
3. Barker, C.: Nominals don't provide criteria od identity. In: Alexiadou, A., Rathert, M. (eds.) Nominalizations across Languages and Frameworks. Intrface Explorations (2008)
4. Beeson, M.: Foundations of Constructive Mathematics. Springer (1985)
5. Bishop, E.: Foundations of Constructive Analysis. McGraw-Hill (1967)

6. Bunt, H.: Mass Terms and Model-Theoretic Semantics. Cambridge University Press (1985)
7. Church, A.: A formulation of the simple theory of types. J. Symbolic Logic 5(1) (1940)
8. Clarke, D.: Mass terms as subjects. Philosophical Studies 21(1/2) (1970)
9. Coquand, T.: Metamathematical investigations of a calculus of constructions. In: Oddifredi, P. (ed.) Logic and Computer Science (1990)
10. Frege, G.: Grundlagen der Arithmetik. Basil Blackwell (1884), (Translation by J. Austin in 1950: The Foundations of Arithmetic)
11. Geach, P.: Reference and Generality. Cornell University Press (1962)
12. Gupta, A.: The Logic of Common Nouns. Yale University Press (1980)
13. Laycock, H.: Some questions of ontology. Philosophical Review 81(1) (1972)
14. Luo, Z.: Computation and Reasoning: A Type Theory for Computer Science. Oxford Univ. Press (1994)
15. Luo, Z.: Coercive subtyping. J. of Logic and Computation 9(1), 105–130 (1999)
16. Luo, Z.: Type-theoretical semantics with coercive subtyping. Semantics and Linguistic Theory 20 (SALT20), Vancouver (2010)
17. Luo, Z.: Contextual Analysis of Word Meanings in Type-Theoretical Semantics. In: Pogodalla, S., Prost, J.-P. (eds.) LACL 2011. LNCS (LNAI), vol. 6736, pp. 159–174. Springer, Heidelberg (2011)
18. Luo, Z.: Type-theoretical semantics with coercive subtyping. Lecture notes at ESSLLI 2011 (for Lexical Semantics, a course taught together with N. Asher), Ljubljana (2011), http://www.cs.rhul.ac.uk/home/zhaohui/ESSLLI11notes.pdf?
19. Martin-Löf, P.: An intuitionistic theory of types: predicative part. In: Rose, H., Shepherdson, J.C. (eds.) Logic Colloquium'73 (1975)
20. Martin-Löf, P.: Intuitionistic Type Theory. Bibliopolis (1984)
21. Montague, R.: Formal Philosophy. Yale University Press (1974)
22. Nicolas, D.: Mass nouns and plural logic. Linguistics and Philosophy 31(2) (2008)
23. Pelletier, F.: On some proposals for the semantics of mass nouns. J. of Philosophical Logic 3 (1974)
24. Quine, W.: Word & Object. MIT Press (1960)
25. Ranta, A.: Type-Theoretical Grammar. Oxford University Press (1994)
26. Strawson, P.: Individuals: An Essay in Descriptive Metaphyscis. Anchor Books (1963)
27. Sundholm, G.: Constructive generalized quantifiers. Synthese 79(1) (1989)
28. Werner, B.: On the strength of proof-irrelevant type theories. Logical Methods in Computer Science 4(3) (2008)
29. Zimmerman, D.: Theories of masses and problems of constitution. Philosophical Review 104(1) (1995)

A Σ-Types

A Σ-type is an inductive type of dependent pairs. Here are the informal descriptions of the basic laws governing Σ-types (see, for example, [20] for the formal rules).

- 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.
- $\Sigma(A, B)$ consists of pairs (a, b) such that a is of type A and b is of type $B(a)$.
- Σ-types are associated 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 $\Sigma(A, B)$.

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 non-dependent pairs.

B Finite Types of Journeys

Finite types are inductive types which contain finitely many objects. They were considered in [19,20] for natural numbers: intuitively, N_n contains the natural numbers $0, 1, ..., n-1$. Here, we introduce them as containing finitely many journeys.

Let T be the type of the journeys in concern. Then, for $n \in \omega$ and $t_i : T$ $(i = 1, ..., n)$, $Fin[t_1, ..., t_n]$ is the finite type with the journeys t_i as its objects. For instance, $Fin[t_1, t_2, t_3]$ contains only three journeys t_1, t_2 and t_3, while $Fin[]$ is an empty type that does not contain any object. Formally, these finite types are specified by means of the rules in Figure 2.

Now, for any $p : Person$, $Journey[T](p)$ is the finite type of journeys in T that p has made. For instance, intuitively, if $Journey[T](p) = Fin[t_1, t_2, t_3]$, p has made journeys t_i $(i = 1, 2, 3)$, while if $Journey[T](p) = Fin[]$, p has made no journeys.[11]

Formation Rule

$$\frac{t_i : T \quad (i = 1, ..., n)}{Fin[t_1, ..., t_n] : Type} \quad (n \in \omega)$$

Introduction Rule

$$\frac{Fin[t_1, ..., t_n] : Type}{t_i : Fin[t_1, ..., t_n]} \quad (i = 1, ..., n)$$

Elimination Rule

$$\frac{c : Fin[t_1, ..., t_n] \quad C : (Fin[t_1, ..., t_n])Type \quad c_j : C(t_j) \ (j = 1, ..., n)}{\mathcal{E}_n(C, c_1, ..., c_n, c) : C(c)}$$

Computation Rule

$$\frac{C : (Fin[t_1, ..., t_n])Type \quad c_j : C(t_j) \ (j = 1, ..., n)}{\mathcal{E}_n(C, c_1, ..., c_n, t_i) = c_i : C(t_i)} \quad (i = 1, ..., n)$$

Fig. 2. Rules for finite types of journeys

[11] The details of the formal definition of $Journey[T]$ is not given here: to do it properly, we need to consider the type structure of $Person$ (eg, its inductive structure) and employs a type universe U that contains the finite types of journeys as objects and to define $Journey[T]$ to be of type $Person \rightarrow U$.