Dependent Event Types

Zhaohui Luo[1]* and Sergei Soloviev[2]

[1] Royal Holloway, Univ of London, U.K.
zhaohui.luo@hotmail.co.uk
[2] IRIT, Toulouse, France
Sergei.Soloviev@irit.fr

Abstract. This paper studies how dependent types can be employed for a refined treatment of event types, offering a nice improvement to Davidson's event semantics. We consider dependent event types indexed by thematic roles and illustrate how, in the presence of refined event types, subtyping plays an essential role in semantic interpretations. It is shown that dependent event types give a natural solution to the incompatibility problem (sometimes called the event quantification problem) in combining event semantics with the traditional Montague semantics.

1 Introduction

The event semantics, whose study was initiated by Davidson [4] and further studied in its neo-Davidsonian turn (see [10] among others), has several notable advantages including Davidson's original motive to provide a satisfactory semantics for adverbial modifications. Dependent types, as those found in Modern Type Theories, provide a useful tool in formalising event types and provide a nice treatment of the event semantics.

An event type may depend on thematic roles such as agents and patients of the events in the type. For example, we can consider the type $Evt_{AP}(a,p)$ of events whose agent and patient are a and p, respectively. It is shown that such dependent event types give a natural solution to the incompatibility problem in combining event semantics with the traditional Montague semantics [2, 13] (sometimes called the event quantification problem [5]): the correct semantics are accepted while the incorrect ones are excluded by typing because they would be ill-typed and hence illegal.

We shall investigate subtyping relations between event types which include dependent types such as $Evt(a,p)$ and the non-dependent type $Event$ of all events (the latter is found in the traditional setting). For example, it may be natural to have $Evt_{AP}(a,p) \leq Evt_A(a)$, that is, the type of events with agent a and patient p is a subtype of that with agent a. With such subtyping relations in place, the semantics of verb phrases can now take the usual non-dependent types, as in the traditional setting, although dependent event types are considered.

* This work is partially supported by EU COST Action CA15123 and CAS/SAFEA International Partnership Program.

Dependent event types were first considered in an example in [1] to study linguistic coercions in formal semantics, where types of events are indexed by their agents: $Evt(h)$ is the type of events conducted by $h : Human$. In this short paper, we shall study event types dependent on thematic roles in a general setting in formal semantics with events both in the traditional Montague semantics [9] and in formal semantics in modern type theories [11, 7]. In §2, we shall describe the basics of dependent event types, introducing notations and examples. Subtyping between event types is described in §3, where we show, for example, how VPs can take the traditional non-dependent type, while we consider dependent event types. §4 shows that a natural solution to the event quantification problem can be found in the setting with dependent event types, giving an example of advantages over the traditional setting. In the concluding section, we shall briefly discuss the future work on dependent event types.

2 Dependent Event Types

In the Davidsonian event semantics in the traditional Montagovian setting [4, 10], there is only one type *Event* of all events. For example, the sentence (1) is interpreted as (2):

(1) John kissed Mary passionately.

(2) $\exists e : Event.\ kiss(e)\ \&\ agent(e, j)\ \&\ patient(e, m)\ \&\ passionate(e)$

where in (2), *Event* is the type of all events, $kiss$, $passionate : Event \rightarrow \mathbf{t}$ are predicates over events, and $agent$, $patient : Event \rightarrow \mathbf{e} \rightarrow \mathbf{t}$ are relations between events and entities.[3] Please note that, in the above neo-Davidson's semantics (2), adverbial modifications and thematic role relations are all propositional conjuncts in parallel with the verb description, an advantageous respect as compared with an interpretation without events.

We propose to consider refined types of events. Rather than a single type *Event* of events, we introduce types of events that are dependent on some parameters $\bar{t} = t_1, ..., t_n$. For instance, an event type can be dependent on agents and patients. Let *Agent* and *Patient* be the types of agents and patients, respectively. Then, for $a : Agent$ and $p : Patient$, the dependent type

$$Evt_{AP}(a, p)$$

is the type of events whose agents are a and whose patients are p. With such dependent event types, the above sentence (1) can now be interpreted as:[4]

(3) $\exists e : Evt_{AP}(j, m).\ kiss(e)\ \&\ passionate(e)$

[3] In logical formulas describing semantics, people often omit the types of events and entities since there are only one type of events and one type of entities.

[4] Please note here that, for $kiss(e)$ and $passionate(e)$ to be well-typed, the type of event e must be the same as the domain of $kiss$ and $passionate$ – see the next section about subtyping, which allows them to be well-typed.

Note that, besides other things we are going to explain below, we do not need to consider the relations *agent* and *patient* as found in (2) because they can now be 'recovered' from typing. For example, for the event types dependent on agents and patients, we can define functions $\textsc{agent}_{AP}$ and $\textsc{patient}_{AP}$ such that, for any event $e : Evt_{AP}(a,p)$, $\textsc{agent}_{AP}(e) = a$ and $\textsc{patient}_{AP}(e) = p$.[5]

The parameters of dependent event types are usually names of thematic roles such as agents and patients. Formally, the dependent event types are parameterised by objects of types $A_1, ..., A_n$. Event types with n parameters are called n-ary event types. In this paper, we shall only consider n-ary event types with $n = 0, 1, 2$:

– When $n = 0$, the event type, usually written as *Event*, has no parameters. *Event* corresponds to the type of all events in the traditional setting.
– When $n = 1$, we only consider $Evt_A : Agent \rightarrow Type$ and $Evt_P : Patient \rightarrow Type$, i.e., the event types dependent on agents and those dependent on patients. For example, if John is an agent with interpretation j, $Evt_A(j)$ is the type of events whose agent is John.
– When $n = 2$, we only consider $Evt_{AP} : Agent \rightarrow Patient \rightarrow Type$, i.e., event types dependent on agents and patients. For example, if John is an agent and Mary is a patient, $Evt_{AP}(j,m)$ is the type of events whose agent and patient are John and Mary, respectively (cf., the example (3) above).

Introducing dependent event types has several advantages. In this paper, we shall detail one of them, that is, it gives a natural solution to the event quantification problem – see §4. Before doing that, we shall consider the subtyping relationship between event types which, among other things, simplifies the semantic interpretations of VPs in the semantics with dependent event types.

3 Subtyping between Event Types

Event types have natural subtyping relationships between them. For example, an event whose agent is a and patient is p is an event with agent a. In other words, for $a : Agent$ and $p : Patient$, the type $Evt_{AP}(a,p)$ is a subtype of $Evt_A(a)$.

[5] Formally, we have

$$\begin{aligned}\textsc{agent}_{AP} &= \lambda a{:}Agent\lambda p{:}Patient\lambda e{:}Evt_{AP}(a,p).a \\ &: \Pi a{:}Agent\Pi p{:}Patient.\ (Evt_{AP}(a,p) \rightarrow Agent)\end{aligned}$$

and similarly, for example,

$$\begin{aligned}\textsc{agent}_{A} &= \lambda a{:}Agent\lambda e{:}Evt_{A}(a).a \\ &: \Pi a{:}Agent.\ (Evt_{A}(a) \rightarrow Agent)\end{aligned}$$

Usually we simply write, for example, $\textsc{agent}_{AP}(e)$ for $\textsc{agent}_{AP}(a,p,e)$ because, in a proof assistant such as Coq, a and p can be automatedly inferred from $e : Evt_{AP}(a,p)$.

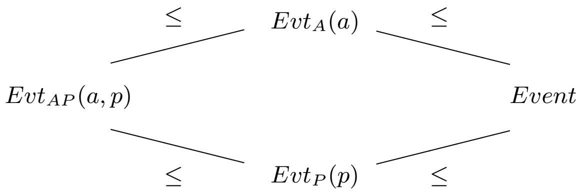

Fig. 1. Subtyping between event types with $a : Agent$ and $p : Patient$.

If we only consider the event types $Event$, $Evt_A(a)$, $Evt_P(p)$ and $Evt_{AP}(a, p)$ (cf., the last section), they have the following subtypnig relationships:

$$Evt_{AP}(a, p) \leq Evt_A(a) \leq Event$$

$$Evt_{AP}(a, p) \leq Evt_P(p) \leq Event$$

which can be depicted as Figure 1.

Formally, the subtyping relationship obeys the following rule (called subsumption rule):

$$(*) \qquad \frac{a : A \quad A \leq B}{a : B}$$

It is also reflexive and transitive. The underlying type theory for formal semantics can be extended by dependent event types together with the subtyping relations governed by the subsumption rule.[6]

The incorporation of subtyping between event types is not only natural but plays an essential role in semantic interpretations. This can best be explained by considering how verb phrases are interpreted. In the neo-Davidson's event semantics (with only $Event$ as the type of events), a verb phrase is interpreted as a predicate over events, as the following example shows.

(4) $talk : Event \rightarrow \mathbf{t}$.

(5) John talked loudly.

(6) $\exists e : Event.\ talk(e)\ \&\ loud(e)\ \&\ agent(e, j)$

With refined event types such as $Evt_A(j)$, how can we interpret *talk* and (5)? In analogy, the desired semantics of (5) would be (7), where the agent of the event

[6] An underlying type theory can either be the simple type theory [3] in the Montagovian semantics or a Modern Type Theory in MTT-semantics as considered in, for example, [7]. We also mention that, in the setting of MTT-semantics, coercive subtyping [6, 8] is used and, for uniformity, we may adopt coercive subtyping rather than subsumptive subtyping in the MTT-seting, although in general subsumptive subtyping is simpler.

e can be obtained as $\textsc{agent}_A(e) = j$:

(7) $\exists e : Evt_A(j).\ talk(e)\ \&\ loud(e)$

However, if *talk* is of type $Event \to \mathbf{t}$, $talk(e)$ would be ill-typed since e is of type $Evt_A(j)$, not of type *Event*. Is (7) well-typed? The answer is, if we do not have subtyping, it is not. But, if we have subtyping as described above, it is! To elaborate, because $e : Evt_A(j) \leq Event$, $talk(e)$ is well-typed by the subsumption rule ($*$). Similarly, we have $loud : Event \to \mathbf{t}$ and, therefore, $loud(e)$ is well-typed for $e : Evt_A(j) \leq Event$ as well.

To summarise, the subtyping relations have greatly simplified the event semantics in the presence of refined dependent event types.

Remark 1. Two remarks should be made briefly.

- In this paper, we have mainly illustrated how to extend the Montagovian setting (simple type theory) with dependent event types and their subtyping relations. It is worth remarking that the same applies to modern type theories when they are used for formal semantics (the MTT-semantics as considered in, for example, [7]).
- The subtyping relations also facilitate a natural relationship between the functions such as $\textsc{agent}_{AP}$ and $\textsc{agent}_A$ (see §2 and Footnote 5). For example, because of subtyping relations as depicted in Fig 1, for $e : Evt_{AP}(a, p)$, the following holds by definition:

$$\textsc{agent}_{AP}(a, p, e) = \textsc{agent}_A(a, e) = a$$

because $Evt_{AP}(a, p) \leq Evt_A(a)$.

4 Event Quantification Problem

It has been argued that there is some incompatibility between (neo-)Davidsonian event semantics and the traditional compositional semantics [2, 13]. De Groote and Winter [5] have dubbed this as the *event quantification problem*. Consider the following sentence (8) which, under the traditional event semantics, could have two possible interpretations (9) and (10), where (10) is incorrect.

(8) No dog barks.

(9) $\neg\exists x : \mathbf{e}.\ dog(x)\ \&\ \exists e{:}Event.\ bark(e)\ \&\ agent(e, x)$

(10) (#) $\exists e : Event.\ \neg\exists x : \mathbf{e}.\ dog(x)\ \&\ bark(e)\ \&\ agent(e, x)$

Formally, the incorrect interpretation is acceptable just as the correct one: (10) is a legal formula. In order to avoid such incorrect interpretations as (10), people have made several proposals (see, for example, [2, 13]) which involve, for instance, consideration of quantification not over events but over sets of events [2], or some (informal and somewhat *ad hoc*) principles whose adherence would disallow the incorrect interpretations.

In our setting with dependent event types, this problem is solved naturally and *formally* – the incorrect semantic interpretations are excluded because they ill-typed (in the empty context, where semantic interpretations of whole sentences like (8) are considered). For example, (8) will be interpreted as (11), while the 'incorrect' interpretation (12) is not available (the formula (12) is ill-typed because x in $Evt_A(x)$, outside the scope of second/bound x (although intuitively it refers to it), is a free variable without being declared.)

(11) $\neg\exists x : \mathbf{e}.\ (dog(x)\ \&\ \exists e : Evt_A(x).\ bark(e))$

(12) (#) $\exists e : Evt_A(x).\ \neg\exists x : \mathbf{e}.\ dog(x)\ \&\ bark(e)$

This offers a natural solution to the event quantification problem. Compared with existing solutions with informal ad hoc principles such as those mentioned above, our solution comes naturally as a 'side effect' of introducing dependent event types: it is formally disciplined and natural.

5 Conclusion and Further Investigations

In this paper, we have introduced dependent event types for formal semantics. Subtyping is shown to play an essential role in this setting. We have also considered how dependent event types naturally solve the event quantification problem in combining event semantics with the traditional compositional semantics.

The paper reports the initial findings on this topic of dependent event types, which provide us with a framework that offers potentially new promising insights. The notion of event types as studied in this paper is *intensional*, rather than extensional. For instance, when considering inverse verb pairs such as `buy` and `sell`, one may think that the events in (13) and (14) are the same event [12]; i.e., they are extensionally the same, but intensionally different.

(13) John bought the book from Mary.

(14) Mary sold the book to John.

Work need be done to study the relationship between intensional and extensional events and relevant inference patterns. More generally, as future work, we shall investigate how to understand the sameness of events in the setting with dependent event types.

Another interesting research topic is to study whether general thematic roles should be considered as parameters of event types. Unlike Agent and Patient, some thematic roles considered in the literature may not be suitable to play the role of indexing dependent event types. In such cases, we would tentatively propose that they should still be formalised by means of predicates. However, more careful studies are called for.

Acknowledgement. Thanks go to David Corfield who has spotted an error in a previous version and given useful suggestions, and to Christian Retoré for helpful comments on this work.

References

1. N. Asher and Z. Luo. Formalisation of coercions in lexical semantics. *Sinn und Bedeutung 17, Paris*, 2012.
2. L. Champollion. The interaction of compositional semantics and event semantics. *Linguistics and Philosophy*, 38:31–66, 2015.
3. A. Church. A formulation of the simple theory of types. *J. Symbolic Logic*, 5(1), 1940.
4. D. Davidson. The logical form of action sentences. *In: S. Rothstein (ed.). The Logic of Decision and Action. University of Pittsburgh Press*, 1967.
5. P. de Groote and Y. Winter. A type-logical account of quantification in event semantics. *Logic and Engineering of Natural Language Semantics 11*, 2014.
6. Z. Luo. Coercive subtyping in type theory. *CSL'96, LNCS'1258*, 1997.
7. Z. Luo. Formal semantics in modern type theories with coercive subtyping. *Linguistics and Philosophy*, 35(6):491–513, 2012.
8. Z. Luo, S. Soloviev, and T. Xue. Coercive subtyping: theory and implementation. *Information and Computation*, 223:18–42, 2012.
9. R. Montague. *Formal Philosophy.* Yale University Press, 1974. Collected papers edited by R. Thomason.
10. T. Parsons. *Events in the Semantics of English.* MIT Press, 1990.
11. A. Ranta. *Type-Theoretical Grammar.* Oxford University Press, 1994.
12. A. Williams. *Arguments in Syntax and Semantics.* Cambridge, 2015.
13. Y. Winter and J. Zwarts. Event semantics and abstract categorial grammar. *Proc. of Mathematics of Language 12, LNCS 6878*, 2011.