Quantifying over eventualities in continuation semantics

Anca Dinu*

University of Bucharest, Faculty of Foreign Languages
Center for Computational Linguistics

The paper explores the possibilities of representing the semantics of eventualities in continuation semantics framework. It focuses on providing an analysis for quantification over eventualities and for anaphora to eventualities. We give specific lexical entries for the adverbial quantifiers *always* and *never* and for a silent adverbial quantifier we consider responsible for the meaning of expressions with no overt adverbial quantifiers. We point out that quantificational adverbs usually have an implicit restriction (i.e. the contextually relevant eventualities), as opposed to the nominal quantifiers where the restrictor is the noun phrase. This eventuality restriction is a set of eventualities that cannot be completely specified or enumerated. Nevertheless, the human mind has no problem to operate with this kind of vagueness. Also, we argue that the Scope Domain Principle, which says that the eventuality quantifier *always* takes lowest possible scope with respect to other quantifiers, is too strong.

AMS 2010: 03B65, 68T50, 03B65.

Key words: discourse semantics; logical form; continuations; anaphora; eventualities or situations; adverbial quantifiers.

1 INTRODUCTION

We explore in this paper the possibilities of representing the semantics of events (in the sense of [2], [3]) in the continuation semantics framework (as...
presented in [4]). The computer science concept of continuations has been previously used to account for intra-sentential linguistic phenomena such as focus fronting, donkey anaphora, presuppositions, crossover or superiority in [5], [6], [24], [23], [4], for cross-sentential semantics in [14] and for analyzing discourse structure in [1]. The merit of continuations in the context of dynamic semantics is that they abstract away from assignment functions that are essential to the formulations of Dynamic Intensional Logic, Dynamic Montague Grammar, Dynamic Predicate Logic and Discourse Representation Theory, thus do not have problems like the destructive assignment problem in DPL or the variable clash problem in DRT.

The focus of this paper is on providing an analysis for quantification over events and for anaphora to events in continuation semantics framework (we use the term anaphora in its linguistic sense, that is, an instance of an expression referring to another one, usually located in preceding utterances).

In section 2 we present the formalism we use.

In section 3, we give specific lexical entries for the adverbial (eventuality) quantifiers always and never and for a silent adverbial quantifier we consider responsible for the meaning of expressions with no overt adverbial quantifiers. We provide here enough details to make plausible the interpretation of events in continuation semantics framework. We point out that quantificational adverbs usually have an implicit restriction (i.e. the contextually relevant events), as opposed to the nominal quantifiers (such as every, any, no, each, some, etc.) where the restrictor is the noun phrase. This eventuality restriction is a set of eventualities that cannot be completely specified or enumerated. Nevertheless, the human mind has no problem to operate with this kind of vagueness. Also, we argue that the Scope Domain Principle (adapted from [18], cf. [21]), which says that the eventuality quantifier always takes lowest possible scope with respect to other quantifiers, is too strong. Instead, we suggest that the scope behavior of eventuality quantifiers is ambiguous and it is a discourse matter to decide which reading is preferred.

In section 4 we conclude and point out that eventualities semantics needs no extra stipulations to be accounted for in this framework. This is due to the fact that the continuation based semantics provides a unified account of scope-taking. Thus, once we get the scope of the lexical entries right for a particular discourse, we automatically get the right truth conditions and interpretation for that piece of discourse. We also highlight the importance of further research on relevant aspects (such as thematic roles, modality, tense, aspect or distributivity) without which a complete account of event semantics cannot be possible.
Continuations are a standard tool in computer science, used to control side effects of computation (such as evaluation order, printing or passing values). The basic idea of writing a grammar in *Continuation passing style* ("continuizing" a grammar) is to provide subexpressions with direct access to their own continuations (future context), so subexpressions are modified to take a continuation as an argument. *Continuation passing style* is in fact a restricted (typed) form of lambda-calculus.

An undelimited continuation of an expression represents "the entire (default) future for the computation" of that expression. [12] introduced delimited continuations (sometimes called ‘composable’ continuations) such as control (‘C’) and prompt (‘%’). Delimited continuations represent the future of the computation of the expression up to a certain boundary. The event semantics discussed here makes use only of delimited continuations.

For instance, if we take the local context to be restricted to the sentence, when computing the meaning of the sentence *John saw Mary*, the default future of the value denoted by the subject is that it is destined to have the property of seeing Mary predicated of it. In symbols, the continuation of the subject denotation *j* is the function \( \lambda x. \text{saw } m \ x \). Similarly, the default future of the object denotation *m* is the property of being seen by John, the function \( \lambda x. \text{saw } x \ j \); the continuation of the transitive verb denotation *saw* is the function \( \lambda R. R \ m \ j \); and the continuation of the verb phrase *saw Mary* is the function \( \lambda P. P \ j \). This simple example illustrates two important aspects of continuations:

1. the continuation of an expression is always relative to some larger expression containing it;
2. every meaningful subexpression of a larger expression has a continuation.

Thus, when *John* occurs in the sentence *John left yesterday*, *John’s* continuation is the property \( \lambda x. \text{yesterday left } x \); when it occurs in *Mary thought John left*, its continuation is the property \( \lambda x. \text{thought}(\text{left } x) \ m \) and when it occurs in the sentence *Mary or John left*, its continuation is \( \lambda x. (\text{left } m) \lor (\text{left } x) \) and so on.

We will refer to the semantics of a natural language fragment which uses the notion of continuations from the series of articles [5], [6], [24], [23], [4], as *continuation semantics*. We use in this paper its variant as presented in [4], which we shortly recall in the following.
For a given expression we use a tower notation, which consists of three levels: the top level specifies the syntactic category of the expression coached in Categorial Grammar*, the middle level is the expression itself and the bottom level is the semantic value.

\[
\begin{array}{c}
\text{syntactic category} \\
\text{expression} \\
\text{semantic value}
\end{array}
\]

The syntactic categories are written \( \text{C} | \text{B} | \text{A} \), where \( A \), \( B \) and \( C \) can be any syntactic categories. We read this counter clockwise as ”the expression functions as a category \( A \) in local context, takes scope at an expression of category \( B \) to form an expression of category \( C \).”

We use the following denotations of syntactic categories: \( S \) for the sentence, \( VP \) for the verb phrase, \( DP \) for the determiner phrase and \( N \) for the noun.

The semantic value is the denotation (extension) of the expression. Bold typeface is commonly used for denotations: for instance, \( j \) is the denotation (reference) of John, \( \text{see} \) is the denotation of the verb \( \text{see} \) (i.e. a function that assigns to all pairs of entities the truth value one, if the pairs are in \( \text{see} \) relation and to truth value zero, if the pairs are not in \( \text{see} \) relation).

The semantic value \( \lambda x. f[k(x)] \) is equivalently written vertically as \( f[] \) omitting the future context (continuation) \( k \). Here, \( x \) can be any expression, and \( f[] \) can be any expression with a gap \([\)\. Free variables in \( x \) can be bound by binders in \( f[] \). This notational convention is meant to make easier (more visual) then in linear notation the combination process of two expressions: a left expression (left-exp) and a right expression (right-exp). Here are the two

---

* The term Categorial Grammar (CG) names a group of theories of natural language syntax and semantics in which the complexity is moved from rules to lexical entries. Formally, a categorial grammar is a quadruple \( (\Sigma, \text{Cat}, S, :=) \), where \( \Sigma \) is a finite set of symbols, \( \text{Cat} \) is a finite set of primitive categories, \( S \in D(\text{Cat}) \) and the relation \( := \) is the lexicon which relates categories to symbols \( := \subseteq D(\text{Cat}) \times \Sigma \). \( D(\text{Cat}) \) is the least set such that \( \text{Cat} \subseteq D(\text{Cat}) \) and if \( X, Y \in D(\text{Cat}) \) then \( (X/Y), (X\setminus Y) \in D(\text{Cat}) \). \( X/Y \) and \( Y\setminus X \) represent function types: \( X/Y \) is the type of a phrase that results in a phrase of type \( Y \) when followed (on the right) by a phrase of type \( X \); \( X\setminus Y \) is the type of a phrase that results in a phrase of type \( Y \) when preceded (on the left) by a phrase of type \( X \). There are two rules: application \( (X/Y) \) concatenated with \( Y \text{gives} X \) or \( Y \) concatenated with \( X \text{gives} X \) and composition \( (X/Y) \) concatenated with \( Y \text{gives} X/Z \). We refer to [20] for a most recent survey of categorial grammars.
possible modes of combination [4]:

\[
\begin{pmatrix}
\frac{C|D}{A/B} & \frac{D|E}{B} \\
\left( \begin{array}{c}
g[l] \\
\frac{h[x]}{x} \\
\end{array} \right) & \left( \begin{array}{c}
g[l] \\
\frac{h[x]}{x} \\
\end{array} \right)
\end{pmatrix}
= \left( \begin{array}{c}
g[l] \\
\frac{h[x]}{x} \\
\end{array} \right)
\]

\[
\begin{pmatrix}
\frac{C|D}{B} & \frac{D|E}{B|A} \\
\left( \begin{array}{c}
g[l] \\
\frac{h[x]}{x} \\
\end{array} \right) & \left( \begin{array}{c}
g[l] \\
\frac{h[x]}{x} \\
\end{array} \right)
\end{pmatrix}
= \left( \begin{array}{c}
g[l] \\
\frac{h[x]}{x} \\
\end{array} \right)
\]

Below the horizontal lines, combination proceeds simply as in combinatory Categorial Grammar: in the syntax, \( B \) combines with \( A/B \) or \( B \setminus A \) to form \( A \); in the semantics, \( x \) combines with \( f \) to form \( f(x) \). Above the horizontal lines is where the combination machinery for continuations kicks in. The syntax combines the two pairs of categories by a kind of cancellation: the \( D \) on the left cancels with the \( D \) on the right. The semantics combines the two expressions with gaps by a kind of composition: we plug \( h[] \) into the gap of \( g[] \), to form \( g[h][] \). The expression with a gap on the left, \( g[] \), always surrounds the expression with a gap on the right, \( h[] \), no matter which side supplies the function and which side supplies the argument below the lines. This fact expresses the generalization that the default order of semantic evaluation is left-to-right.

When there is no quantification or anaphora involved, a simple sentence like John came is derived as follows:

\[
\begin{pmatrix}
DP & DP\setminus S \\
John & came \\
j & came
\end{pmatrix}
= \begin{pmatrix}
S \\
John came \\
\end{pmatrix}
\]

When there is no quantification or anaphora involved, a simple sentence like John came is derived as follows:

In the syntactic layer, as it is usual in categorial grammar, the category under slash (here \( DP \)) cancels with the category of the argument expression; the semantics is function application.

Quantificational expressions have extra layers on top of their syntactic category and on top of their semantic value, making essential use of the powerful mechanism of continuations in ways proper names or definite descriptions do not. For example, below is the derivation for A man came.

\[
\begin{pmatrix}
\frac{S|S|N}{\lambda P. P(x)\land \frac{\text{man}}{\text{man came}}} \\
\frac{\text{a man came}}{\text{came}} \\
\end{pmatrix}
= \begin{pmatrix}
\frac{\text{a man came}}{\text{came}} \\
\frac{\text{a man came}}{\text{came}} \\
\end{pmatrix}
\]

5
The comparison between the above analysis of *John came* and that of *A man came* reveals that *came* has been given two distinct values. The first, simpler value is the basic lexical entry, the more complex value being derived through the standard type shifter Lift, proposed by [22], [16], [26], and many others:

\[
\frac{A \xrightarrow{\text{expression}} \lambda x. x}{B \xrightarrow{\text{expression}} \lambda k.k \frac{\lambda x.x}{x}} \quad (5)
\]

Syntactically, *Lift* adds a layer with arbitrary (but matching!) syntactic categories. Semantically, it adds a layer with empty brackets. In linear notation we have: \( x \xrightarrow{\text{Lift}} \lambda k.k(x) \). To derive the syntactic category and a semantic value with no horizontal line, Barker and Shan introduce the type-shifter *Lower* in [4]. In general, for any category \( A \), any value \( x \), and any semantic expression \( f[\ ] \) with a gap, the following type-shifter is available.

\[
\frac{A \xrightarrow{\text{expression}} \lambda x. x}{B \xrightarrow{\text{expression}} \lambda k.k \frac{\lambda x.x}{x}} \quad (6)
\]

Syntactically, *Lower* cancels an \( S \) above the line to the right with an \( S \) below the line. Semantically, *Lower* collapses a two-level meaning into a single level by plugging the value \( x \) below the line into the gap [ ] of the expression \( f[\ ] \) above the line. *Lower* is equivalent to identity function application.

The third and the last type shifter we need is the one that treats the binding. Binding is a term used both in logics and in linguistics with analog (but not identical) meaning. In logics, a variable is said to be bound by an operator (as the universal or existential operators) if the variable is inside the scope of the operator. If a variable is not in the scope of any operator, than the variable is said to be free. In linguistics, a binder may be a constituent such as a proper name (*John*), an indefinite common noun (*a book*), an event or a situation. Anaphoric expressions such as pronouns (*he, she, it, him, himself*, etc.), definite common nouns (*the book, the book that John read*, etc.), demonstrative pronouns (*this, that, etc.*), act as variables that take the value of (are bind by) a previous binder. We adopt the idea (in line with [4]) that the mechanism of binding is the same as the mechanism of scope taking.

In order to give a proper account of anaphoric relations in discourse, we need to formulate an explicit semantics for both the binder and the anaphoric expressions to be bound. Any DP may act as a binder, as the Bind Rule from [4] explicitly states:
At the syntactic level, the Bind rule says that an expression that functions in local context as a DP may look to the right to bind an anaphoric expression (Barker and Shan encode that by the sign $\triangleright$). At the semantic level, the expression transmits (copies) the value of the variable $x$. In linear notation, the semantic part of the Bind rule looks like: $\lambda k.f[k(x)] \triangleright\triangleright \lambda k.f[\{k(x)\}]$.

As for the elements that may be bound, the singular pronoun *he* has, for instance, the following lexical entry ([4]):

$$
\begin{array}{c}
\text{DP} \\
\text{S} \\
\text{DP} \\
\text{he} \\
\lambda y. \\
\end{array}
$$

(8)

To account for multiple anaphoric expressions (and their binders) or for inverse scope of multiple quantifiers, each binder can occupy a different scope-taking level in the compositional tower. With access to multiple levels, it is easy to handle multiple binders. Then, a pronoun or another anaphoric expression chooses its binder by choosing where to take scope. So, distinct scope-taking levels correspond to different binders, layers playing the role of indices: a binder and the anaphoric expression it binds must take effect at the same layer in the compositional tower. A superior level takes scope at inferior levels and left expressions take scope at right expressions, to account for left-to-right natural language order of processing.

Dinu [10] introduced the semantics of punctuation mark "." (dot), thus extending continuation semantics from sentence level to discourse level. *Dot* is considered as a function that takes two sentence denotations and returns a sentence denotation (the conjunction of the two sentence denotations):

$${S \setminus (S/S)} \quad {S \setminus (S/S)}$$

(9)

For two affirmative sentences with no anaphoric relations and no quantifiers, such as *John came. Mary left*, the derivation trivially proceeds as follows:
\[ S \quad S\backslash (S/S) \quad S \quad S \]

\[ \text{John came} \quad \text{Mary left} = \text{John came.Mary left} \quad (10) \]

\[ \text{came } j \quad \lambda p \lambda q. p \land q \quad \text{left } m \quad \text{came } j \land \text{left } m \]

As one sees above, there is no need in this simple case to resort to type shifting at all. Nevertheless, type shifting and the powerful mechanism of continuations are employed when dealing with linguistic side effects such as quantification or anaphora. For instance, the derivation of the denotation of *A man came. He whistled* is:

\[
\frac{S|S \quad S}{DP/N} \quad \frac{N}{a\text{ man}} = \frac{S|S}{\lambda x. P(x) \land [\_]} \quad \frac{\lambda x. \text{man}(x) \land [\_]}{\exists x. \text{man}(x) \land [\_]} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\exists x. \text{man}(x) \land [\_]} \quad (11)
\]

\[
\frac{\exists x. \text{man}(x) \land [\_]}{\text{came } x} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{came } x} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{came } x} \quad (12)
\]

\[
\frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad (13)
\]

\[
\frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad (14)
\]

\[
\frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad (15)
\]

\[
\frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad \frac{\exists x. \text{man}(x) \land [\_]}{\text{whistled } y} \quad (16)
\]

\[ S \quad \text{a man came. he whistled} \quad \text{Lower} \quad \frac{\exists x. \text{man}(x) \land (\lambda y.) \text{came } x \land \text{whistled } y] [\_]}{\text{whistled } y} \quad (16) \]

\[ S \quad \text{a man came. he whistled} \quad (16) \]

\[ \exists x. \text{man}(x) \land (\lambda y.) \text{came } x \land \text{whistled } y] [\_] \]
Note that the denotations of came and whistled were also lifted so as to match the ones of a and he, both being scope-takers. The last equality sign is due to routine lambda conversion.

3 ACCOUNTING FOR EVENTUALITIES

In the above formalism we only allowed quantification over entities and anaphora to entities. But natural language expressions may also quantify over and refer to eventualities. We will use the term eventualities in the sense of [2], [3], who proposed the division of eventualities into states, processes (activities) and events (achievements or accomplishments). Although this categorization has important grammaticality implications, we will not use here the distinction between these types of eventualities, considering only the most general type: the eventualities themselves. We will be interested in giving a semantics that copes with reference to such objects. For instance, consider direct reference to events, e.g. event nominalizations (the fall of the Berlin Wall), quantification over events, (John kissed Mary twice. or John always sings.) or anaphoric pronominal reference to eventualities across sentence boundaries, (John kissed Mary. She liked it. / That was nice of him.).

It is worth mentioning here that the notion of event is often used sloppily to mean eventuality, or situation, or sometimes even possible world. Roughly speaking, the difference between those notions is as follow: event is a particular case of eventuality; an eventuality is a situation with a minimality condition included; a situation is a partial possible world. In this paper, we have chosen to use eventualities for simplicity reasons. The semantic types are complex enough anyway, without adding the complexity of situation semantics or of possible worlds. Enriching the class of semantic types may as well be a topic for further research.

Beaver and Condoravdi [8] point that, in Davidsonian Event Semantics ([9], [21], [18], [11] among others), the analysis of quantification is problematic: "either quantifiers are treated externally to the event system and quantified in (cf. [18]), or else the definitions of the quantifiers must be greatly (and non-uniformly) complicated (cf. [17])". Lucas Champollion [7] advocates
for a straightforward analysis of quantification in neo-Davidsonian Event Semantics based on standard type-shifting very much resembling continuations.

In our approach, the eventuality’s ontological status will be that of basic entity of a special type $E$, which we add to the type system that comprises the other two types: the type $e$ for individual entities and the type $t$ for the truth values. We will dub the corresponding category $S_E$ (of type $E$). We will consider predicates (verbs) as functions that take an eventuality as an extra argument. The only thing we might want to be careful about is the order in which a verb takes its arguments: it first takes the (indirect) direct object (if any), then the event variable and, finally, the subject. Thus, a transitive verb for instance will have the form: $\lambda y.\lambda e.\lambda x.\text{verb } y e x$. This order of applying the arguments is meant to mimic the default surface word order in English (John always/never reads a book). The notational difference between individual variables and eventuality variables will be that the former will be notated as usual with $x,y,z$, and the latter with $e, e', e''$. Adverbial quantifiers quantify over sets of events/situations [15], [25], [13]). They do that either overtly (never, always, twice, mostly, usually, often), or covertly by a silent existential adverbial quantifier. We propose that its lexical entry may have the following alternative forms, one for the existential interpretation and one for the universal interpretation, respectively:

$$
\frac{SIS}{S_E}
\frac{\phi_{ev}^e}{\exists x.\text{RelEv}(e) \land [\ ]_e}
\tag{18}
$$

$$
\frac{SIS}{S_E}
\frac{\phi_{ev}^e}{\forall x.\text{RelEv}(e) \rightarrow [\ ]_e}
\tag{19}
$$

The existential silent quantifier is generally used with past tense in constructions such as John came. The universal silent quantifier is generally used with present tense in generic constructions such as John smokes or Birds fly. Although the linguistic notions of time, modality and aspect are of great help in disambiguating existential versus universal event quantification in sentences that lack an overt adverbial quantifier, the ambiguity cannot be resolved always only on such bases. Certain vagueness is implicit in sentences with covert adverbial quantifiers. For instance, in languages with no continuous aspect, a sentence such as John smokes may mean that John always smokes or that there is a smoking event of John at the speech time.

10
We give \textit{always} the following lexical entry:

\[
\frac{S|S}{S_E} \ \
\text{always} \ \
\frac{\forall x. \text{RelEv}(e) \rightarrow [ \cdot ]}{e}
\]  

(20)

We also give the following lexical entry to the adverbial quantifier \textit{never}:

\[
\frac{S|S}{S_E} \ \
\text{never} \ \
\frac{\neg \exists x. \text{RelEv}(e) \land [ \cdot ]}{e}
\]  

(21)

Observe that quantificational adverbs usually have an implicit restriction, (i.e. the contextually relevant situation), as opposed to the nominal quantifiers (such as \textit{every}, \textit{any}, \textit{no}, \textit{each}, \textit{some}, etc) where the restrictor is the noun phrase. Sometimes the restrictor is overtly manifested by means of \textit{if} or \textit{when} clauses, as in \textit{If the door is open, John came}. However, most of the time, the restrictor is implicit. In this case, \text{RelEv} is a set of eventualities that cannot be completely specified or enumerated. Nevertheless, the human mind has no problem to operate with this kind of vagueness. The implementation of the event semantics presented in this paper needs a way of representing the imprecision of the \text{RelEv} restriction. We leave this issue to further research.

Note also that variable \( e \) over eventualities from the above lexical entries is not to be confused with the semantic type \( e \) of entities.

In this framework, \textit{John came} receives the following interpretation (using the existential silent quantifier):

\[
\frac{S|S}{DP} \ \
\frac{S|S}{S_E} \ \
\frac{S|S}{S_E \setminus (DP \setminus S)}
\]

\[
\text{John} \ \
\phi_{ev} \ \
\frac{\exists x. \text{RelEv}(e) \land [ \cdot ]}{e} \ \
\text{came} \ \
\frac{\cdot}{\cdot}
\]

\[
(22)
\]

\[
\frac{S|S}{DP} \ \
\frac{S|S}{DP \setminus S} \ \
\frac{S|S}{S_E}
\]

\[
\text{John} \ \
\text{came} \ \
\frac{\exists x. \text{RelEv}(e) \land [ \cdot ]}{e} \ \
\text{came} \ \
\frac{\cdot}{\cdot}
\]

\[
(23)
\]

\[
\frac{S}{\text{Lower}} \ \
\frac{\exists x. \text{RelEv}(e) \land \text{came} \ e \ j}{\text{John came}}
\]

\[
(24)
\]
which amounts to say that there is a contextually relevant eventuality/situation $e$ of coming which is true of John.

Up to now, the use of the restriction $\text{RelEv}(e)$ for the silent existential adverbial quantifier may have been awkward. However, its use becomes more transparent in deriving the interpretation of $\text{John smokes}$, this time using the universal silent quantifier:

$$
\frac{\text{John}}{S|S} \quad \frac{\phi_{\text{ev}}}{S|S} \quad \frac{\text{smoke}}{S} \quad \frac{\text{came}}{S} \quad \frac{\forall e. \text{RelEv}(e) \to [\ ]}{\ } \quad \frac{\text{John smoke}}{S|S \setminus (\text{DP}\backslash S)}
$$

which means that, for every eventuality $e$, if $e$ is a relevant eventuality (one that might be a smoking eventuality), than $e$ is a smoking true of John.

To account for eventuality/situation anaphora, we should allow the eventuality category $S_E$ to bind, by adding the following binding rule:

$$
\frac{A|B}{S_E} \quad \text{expression} \quad \frac{f([ \ ])}{S_E} \quad \frac{\text{Bind}}{A|E\vdash B} \quad \frac{\text{expression}}{f([ \ ])}
$$

We also give the following lexical entry for the pronoun that:

$$
\frac{\text{SG}\vdash B|A}{S_E} \quad \frac{\text{that}}{S_E} \quad \frac{\lambda e'.[ \ ]}{e'}
$$

In this framework, the interpretation of a discourse with eventuality anaphora such as $\text{John came. That was a surprise}$ is derived as it follows (ignoring the existential force of a surprise, for simplicity):

$$
\frac{S|S}{S_E} \quad \frac{\phi_{\text{ev}}}{S_E} \quad \frac{\exists e. \text{RelEv}(e) \wedge [ \ ]}{e} \quad \frac{\text{Bind}}{S\mid S_E \vdash S} \quad \frac{\phi_{\text{ev}}}{S_E} \quad \frac{\exists e. \text{RelEv}(e) \wedge [ \ ]}{e}
$$

$$
\frac{S_E \vdash S|S}{S_E} \quad \frac{S|S}{S_E} \quad \frac{S|S}{S_E \setminus (S_E \setminus S)} \quad \frac{\text{That}}{\lambda e'.[ \ ]} \quad \frac{\phi_{\text{ev}}}{\text{SG}} \quad \frac{\exists e'. \text{RelEv}(e') \wedge [ \ ]}{e'} \quad \frac{\text{was a surprise}}{\text{was a surprise}} \quad \frac{\text{was a surprise}}{(29)}
$$
That $\phi$ was a surprise

\[ \lambda e'. [ ] \exists e''. \text{RelEv}(e'') \land [ ] \] was a surprise

\[ S_E \vdash S | S \]

\[ \text{That was a surprise } e'' = (30) \]

\[ S_E \vdash S | S \]

\[ \text{That was a surprise } e'' \]

\[ S_E \vdash S | S \]

\[ \text{That was a surprise } e'' \]

\[ S_E \vdash S | S \]

\[ \text{That was a surprise } e'' \]

\[ \text{John came. That was a surprise } e'' \]

\[ S_E \vdash S | S \]

\[ \exists e. \text{RelEv}(e) \land [ \lambda e'. (\exists e''. \text{RelEv}(e'') \land [ \text{came } e \land \text{was a surprise } e'' ] ) ] \]

\[ S_E \vdash S | S \]

\[ \text{John came. That was a surprise } e'' \]

\[ S_E \vdash S | S \]

\[ \exists e. \text{RelEv}(e) \land [ \lambda e'. (\exists e''. \text{RelEv}(e'') \land [ \text{came } e \land \text{was a surprise } e'' ] ) ] \]

\[ S_E \vdash S | S \]

\[ \text{John always sings} \]

which means that there is a coming event true of John and a surprising event true of John’s coming.

Observe that the category of the verb is a surprise is here $S_E \setminus (S_E \setminus S)$, that is, the subject is of category $S_E$ (an eventuality/situation). This appears to be a complication at first sight, because it means verbs may be polymorphic: is a surprise may also be of category $S_E \setminus (DP \setminus S)$, like, for instance in *The party was a surprise*. But it only means that verbs may take as arguments both regular DPs (of type $e$) and event nominalizations (of type $E$).

The derivation of discourses with overt adverbial quantifiers such as John always sings proceeds similarly.
John always sing \[
∀. RelEv(e) \rightarrow \text{sing}
\]
\[
\text{sing} = (36)
\]

which means that for every eventuality \(e\), if \(e\) is a contextually relevant eventuality (one in which John might sing), than \(e\) is a singing true of John.

There has been argued that (adapted from [18], cf. [21]) the eventuality quantifier always takes lowest possible scope with respect to other quantifiers (Scope Domain Principle). This claim involved examples of expressions containing other scope taking lexical entries that disallow the inverse scope interpretation (the one in which the eventuality quantifier takes wider scope). For instance, in examples such as \(\text{No dog barks, or Spot does not bark}\), the direct scope interpretation is the only possible one:

\[
\text{No dog bark} \quad \text{No dog bark}
\]
\[
\neg \exists x. (\text{dog}(x) \land \text{bark} e) \quad \neg \exists x. (\text{dog}(x) \land \exists e. \text{RelEv}(e) \land \text{bark} e x)
\]
\[
\iff
\]
\[
\text{bark } e
\]
\[
\text{bark } e x
\]
\[
(37)
\]
\[
(38)
\]
\[
(39)
\]
\[
(40)
\]

meaning that there is no dog for which there is a barking event that is done by that dog.

Similarly, the interpretation of \(\text{Spot does not bark}\) is:

\[
\text{Spot not bark}
\]
\[
\neg \exists e. \text{RelEv}(e) \land \text{bark } e
\]
\[
(41)
\]
meaning that there is no relevant eventuality of barking by Spot.

We consider that it is premature to conclude that eventuality quantifiers always take lowest possible scope with respect to other quantifiers based only on this kind of examples. Rather, in these examples, the inverse scope is ruled out by factors such as the use of a particular aspect (perfect aspect, in this case). For instance, the inverse scope interpretation of *No dog barks* is impossible because in English it would be realized using a different aspect (continue aspect): *No dog is barking*, having the following derivation:

\[
\frac{\text{No dog is barking}}{\exists e. RelEv(e) \land \text{bark } e}
\]

meaning that there is a relevant event for which there is no dog that makes true that event of barking.

A stronger motive for rejecting Scope Domain Principle hypothesis is that there are cases where the preferred meaning is the one with wide eventuality quantifier scope, for instance as in *A diplomat always smiles*.

The direct scope interpretation (not preferred) is:

\[
\frac{S|S}{DP} \quad \frac{S|S}{DP|S}
\]

\[a \text{ diplomat always smiles} \]

meaning that there is a relevant event for which there is no dog that makes true that event of barking.
a diplomat always smiles
\[\exists x. \text{diplomat}(x) \land \forall e. (\text{RelEv}(e) \rightarrow \text{smile} e x)\]
meaning that there is a certain diplomat who smiles in all relevant situations.

The inverse scope interpretation (preferred) is:

\[
\frac{\text{S}|S}{\text{S}}
\]

a diplomat always smiles
\[\exists x. \text{diplomat}(x) \land \forall e. (\text{RelEv}(e) \rightarrow \text{smile} e x)\]
meaning that in every relevant eventuality, a diplomat smiles.

\section*{4 CONCLUSIONS AND FURTHER WORK}

Our starting point was [4] continuation semantics. We shifted from sentential level to discourse level and from quantifying over entities and truth values
to quantifying over entities, truth values and eventualities. The key point of
this paper was accounting for quantification over events and for anaphora to
events. We gave specific lexical entries for the adverbial quantifiers *always*
and *never* and for a silent adverbial quantifier which we consider responsible
for the meaning of expressions with no overt adverbial quantifiers. We point
out that quantificational adverbs usually have an implicit restriction, (i.e. the
contextually relevant events), as opposed to the nominal quantifiers where the
restrictor is the noun phrase. This eventuality restriction is a set of eventuali-
ties that cannot be completely specified or enumerated. Nevertheless, the hu-
man mind has no problem to operate with this kind of vagueness. We argued
that the Scope Domain Principle, which says that the eventuality quantifier
always takes lowest possible scope with respect to other quantifiers, is too
strong. Instead, we suggest that the scope behavior of eventuality quantifiers
is ambiguous and it is a discourse matter to decide which reading is preferred.

Event semantics needs no extra stipulations to be accounted for in this
framework. This is due to the fact that the continuation based semantics pro-
vides a unified account of scope-taking. No other theory to our knowledge
lets indefinites, other quantifiers, pronouns and other anaphora interact in a
uniform system of scope taking, in which quantification and binding employ
the same mechanism. Thus, once we get the scope of the lexical entries right
for a particular discourse, we automatically get the right truth conditions and
interpretation for that piece of discourse.

A word about variable renaming is in order here: throughout the examples
in this paper we have conveniently chosen the names of the variables as to be
distinct. Because there are no free variables in the theory, there is no danger
of accidentally binding a free variable. As for the bound variable, the simple
rule is that the current bounded variable is renamed with a fresh variable name
(cf. Barendregt’s *variable convention*) so as all bound variables have distinct
variable names.

We only provided here enough details to make plausible the interpretation
of events in continuation semantics framework, leaving for further research
important issues such as:

1. A complete specification of event semantics, that is obviously not pos-
sible without taking into consideration thematic roles, aspect, modality
and tense.

2. Multiple eventualities as in *All diplomats smile*; for instance, the inter-
pretation:

\[
\frac{\phi_{\text{ev}}}{\exists X. (X = \text{arg max}_X [X: \text{diplomat}(X)]) \land \forall e. \text{RelEv}(e) \rightarrow \text{smile} e}
\]

\[
\frac{\text{all diplomats}}{\exists X. (X = \text{arg max}_X [X: \text{diplomat}(X)]) \land \forall e. \text{RelEv}(e) \rightarrow \text{smile} e X}
\]

\[
\frac{\text{all diplomats smile}}{\forall e. \text{RelEv}(e) \rightarrow \text{smile} e X}
\]

\[
\frac{\text{all diplomats smile}}{\exists X. (X = \text{arg max}_X [X: \text{diplomat}(X)]) \land \forall e. \text{RelEv}(e) \rightarrow \text{smile} e X}
\]

is not a valid interpretation, because we need to distribute multiple events of smiling over the set of diplomats.

REFERENCES


