Dialogue Modelling and the Remit
of Core Grammar

Eleni Gregoromichelaki[†], Yo Sato[‡], Ruth Kempson[†]
Andrew Gargett[†], Christine Howes[‡]
[†]King's College London, [‡]Queen Mary University of London

1 Introduction

In confronting the challenge of providing formal models of dialogue, with its plethora of fragments and rich variation in modes of context-dependent construal, it might seem that linguists face two types of methodological choice: either (a) conversation employs dialogue-specific mechanisms, for which a grammar specific to such activity must be constructed; or (b) variation arises due to independent parsing/production systems which invoke a process-neutral grammar. However, as dialogue research continues to develop, there are intermediate possibilities, and in this paper we discuss the approach developed within *Dynamic Syntax* (*DS*, Kempson et al. 2001, Cann et al. 2005), a grammar framework within which, not only the parser, but indeed "syntax" itself are just a single mechanism allowing the progressive construction of semantic representations in context. Here we take as a case study the set of phenomena classifiable as *clarifications*, *reformulations*, *fragment requests* and *corrections* accompanied by *extensions*, and argue that though these may seem to be uniquely constitutive of dialogue, they are grounded in the mechanisms of *apposition* equivalently usable in monologue for presenting *reformulations*, *extensions*, *self-corrections* etc.

2 Background

The data we focus on are non-repetitive fragment forms of acknowledgements, clarifications and corrections (henceforth, A female, B male):

(1)　　A:　Bob left.
　　　　B:　(Yeah,) the accounts guy.

(2)

　　　　A:　They X-rayed me, and took a urine sample, took a blood sample. Er, the doctor
　　　　B:　Chorlton?
　　　　A:　Chorlton, mhm, he examined me, erm, he, he said now they were on about a slight [shadow] on my heart.

(3)　　A:　Bob left.
　　　　B:　Rob?
　　　　A:　(No,) (Bob,) the accounts guy.

Even though in the literature the fragments in (2)-(3) might be characterised as illustrating distinct construction-types, in our view, they all illustrate how speakers and hearers may contribute, in some sense to be made precise, to the joint enterprise of establishing some shared communicative content, in what might be loosely called *split utterances*. Even (1), an *acknowledgement*, can be seen this way upon analysis: B's addition is similar in structure to an afterthought *extension* that might have been added by A herself to A's fully sentential utterance. It can be seen in (2) that such joint construction of content can proceed incrementally: the *clarification request* in the form of a *reformulation* is provided by B and resolved by A within the construction of a single proposition. In (3) the fragment reply can be taken to involve *correction*, in the sense that, according to the DS analysis of B's fragment question, he has provided content construable as equivalent to that derived by processing *Rob left?* (see Kempson et al. (2007)). Nevertheless such corrections can also incorporate *extensions* in the above sense, enabling a single conjoined propositional content to be derived in a single step.

It might seem that such illustration of diversity of fragment usage is ample evidence of the need for conversation-specific rules. Indeed, Fernández (2006) presents a thorough taxonomy, as well as detailed formal modelling of *Non-Sentential Utterances* (NSUs), referring to contributions such as (1) as *repeated acknowledgements* involving *reformulation*. Fernández models such constructions via type-specific "accommodation rules" which make a constituent of the antecedent utterance "topical". The semantic effect of acknowledgement is then derived by applying an appropriately defined utterance type for such fragments to the newly constructed context. A distinct form of contextual accommodation is employed to model so-called *helpful rejection* fragments, as in (3) (without the reformulation), whereby a *wh*-question is accommodated in the context by abstracting over the content of one of the sub-constituents of the previous utterance. The content of the correction is derived by applying this *wh*-question in the context to the content of the fragment (see also Schlangen (2003) for another classification and analysis).

In contrast, the alternative explored here is whether phenomena such as (1)-(2), both of which are non-repetitive next-speaker contributions, can be handled uniformly using the mechanisms for structure-building made available in the core grammar, without recourse to construction-specific extensions of that grammar and contextual accommodation rules. This is because, in our view, the range of interpretations these fragments receive in actual dialogue seem to form continua with no well-defined boundaries and mixing of functions (see also comments in Schlangen (2003)). Thus we

propose that the grammar itself simply provides mechanisms for processing/integrating such fragments in the current structure while their precise contribution to the interaction can be calculated by pragmatic inferencing if needed (as in e.g. Schlangen (2003)) or, as seems most often to be the case, be left underspecified without disruption to the dialogue.

One bonus of the stance taken here is the promise it offers for elucidating the grammar-parser contribution to the disambiguation task. Part of the challenge of modelling dialogue is the apparent multiplicity of interpretive and structural options opened up during processing by the recurrent, often overlapping fragments as seen in (2) above. Thus, it might seem that the rich array of elliptical fragments available in dialogue adds to its complexity. However, an alternative point of view is to see such phenomena as providing a window on how interlocutors exploit the incrementality afforded by the grammar. The reliance of fragments on context for interpretation, when employed incrementally, enables the hearer to immediately respond to a previous utterance at any relevant point, in a constrained manner, without 'recovering' a propositional unit. Three features of the *Dynamic Syntax* model of dialogue (Purver et al. (2006)), presented below, provide the flexible control required for such processing: (a) word-by-word incrementality (b) interaction with contextually provided information at every step of the construction process (c) tight coordination of parsing and production.
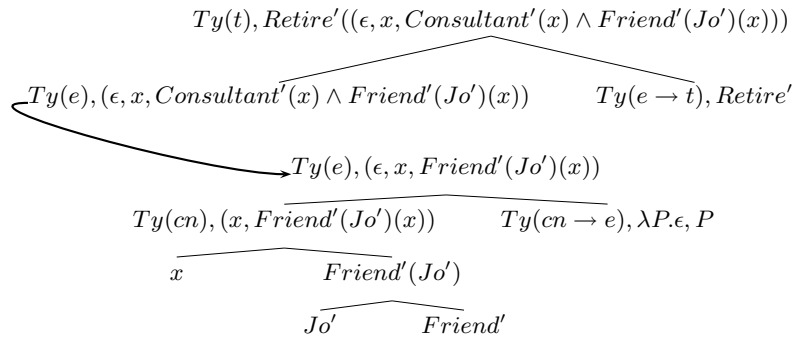
3 Dynamic Syntax: A Sketch

Dynamic Syntax (*DS*) is a parsing-based framework, involving strictly sequential interpretation of linguistic strings. The model is implemented via goal-directed growth of tree structures and their annotations formalised using *LOFT* (Blackburn and Meyer-Viol (1994)), with modal operators $\langle\uparrow\rangle, \langle\downarrow\rangle$ to define concepts of *mother* and *daughter*, and their iterated counterparts, $\langle\uparrow_*\rangle, \langle\downarrow_*\rangle$, to define the notions *be dominated by* and *dominate*. *Underspecification* and update are core aspects of the grammar itself and involve strictly monotonic information growth for any dimension of tree structures and annotations. Underspecification is employed at all levels of tree relations (mother, daughter etc.), as well as formulae and type values, each having an associated *requirement* that drives the goal-directed process of update. For example, an underspecified subject node of a tree may have a requirement expressed in DS with the node annotation $?Ty(e)$, for which the only legitimate updates are logical expressions of type entity ($Ty(e)$); but requirements may also take a modal form, e.g. $?\langle\uparrow\rangle Ty(e \rightarrow t)$, a constraint that the mother node be annotated with a formula of predicate type. Requirements are essential to the dynamics informing the DS account: all requirements must be satisfied if the construction process is to lead to a successful outcome.

Semantic structure is built from lexical and general computational actions. *Computational actions* govern general tree-constructional processes, such as introducing and updating nodes, as well as compiling interpretation for all non-terminal nodes in the tree. Construction of only weakly specified tree relations (*unfixed nodes*) can also be induced, characterised only as dominance by some current node, with subsequent update required. Individual lexical items also provide procedures for building structure in the form of *lexical actions*, inducing both nodes and annotations. Thus *partial trees* grow incrementally, driven by procedures associated with particular words as they are encountered, with a *pointer*, $\diamondsuit$, recording the parser's progress (unlike van Leusen and Muskens (2003), partial trees are part of the model and, unlike in other frameworks, incrementality is word-by-word rather than sentence-by-sentence).

Complete individual trees are taken to correspond to predicate-argument structures (with an event term associated with tense, suppressed in this paper). The *epsilon calculus* (see e.g. Meyer-Viol (1995)) provides the semantic representation language. Complex structures are obtained via a general tree-adjunction operation licensing the construction of so-called *linked* trees, hosting information that is eventually transferred onto the tree FROM which the link is made (Kempson et al 2001). Structures projected as such paired trees range over restrictive relatives, nonrestrictive relatives, conditionals, topic structures and appositions as here. As the semantic representations employ the *epsilon calculus*, eventual compound *epsilon terms* (e.g. $\epsilon, x, P(x)$) are constructed incrementally through *link-adjunction*:

(4) A consultant, a friend of Jo's, is retiring

$$Ty(t), Retire'((\epsilon, x, Consultant'(x) \wedge Friend'(Jo')(x)))$$

$$Ty(e), (\epsilon, x, Consultant'(x) \wedge Friend'(Jo')(x)) \qquad Ty(e \rightarrow t), Retire'$$

$$Ty(e), (\epsilon, x, Friend'(Jo')(x))$$

$$Ty(cn), (x, Friend'(Jo')(x)) \qquad Ty(cn \rightarrow e), \lambda P.\epsilon, P$$

$$x \qquad Friend'(Jo')$$

$$Jo' \qquad Friend'$$

Underspecification of content as well as structure are central to facilitating successful linguistic interaction, our primary concern here. Pronouns, the prototypical case, contribute a place-holding metavariable, noted as e.g. $\mathbf{U}$, plus an associated requirement for update by an appropriate term value: $?\exists \mathbf{x}.Fo(\mathbf{x})$. Equally, definite NPs contribute place-holders plus a constraint providing a restriction/"presupposition" on the kind of entity picked out, e.g., *the man* contributes the annotation $\mathbf{U}_{Man'(\mathbf{U})}, Ty(e)$. The subscript specification is shorthand for a transition to a *linked* tree whose root node is

131

annotated with a formula $Man'(\mathbf{U})$[1]. The update of metavariables can be accomplished if the context contains an appropriate term for substitution: *context* involves storage of *parse states*, i.e., storing of partial tree, word sequence to date, plus the actions used in building up the partial tree (Purver et al 2006).

Scope dependencies between constructed terms or the index of evaluation (e.g. S) are defined on completed propositional formulae, relative to incrementally collected scope constraints (of the form $x < y$ for constructed terms containing variables x and y respectively). Constraints reflect on-line processing considerations modulo over-riding lexical stipulations. For example, proper names contribute as *iota terms*, i.e, epsilon terms reflecting uniqueness in the context, $\iota, x, Bob'(x)$, and these project a scope dependency solely on the index of evaluation reflecting their widest scope property (cf Kamp and Reyle 1994). The structure projected from A's utterance in (1) is thus (5) (note that trees are the RESULT of processing words but do not encode the structure of strings, word order etc., only semantic content derived in interaction with context, thus are the equivalents of DRSs):

$$(5) \quad \begin{array}{c} S < x\ Ty(t), Leave'((\iota, x, Bob'(x))) \\ \overbrace{} \\ Ty(e),(\iota, x, Bob'(x)) \qquad Ty(e \to t), Leave' \end{array}$$

The *scope evaluation* rule reflects the predicate-logic/epsilon-calculus equivalence $\exists x F(x) \equiv F(\epsilon, x, F(x))$ so evaluated terms eventually reflect their containing structure. Hence, evaluation of (5) yields:

$$(6) \quad Ty(t), Leave'(\iota, x, Bob'(x) \wedge Leave'(x))$$

A major aspect of the DS dialogue model is that both *generation* and *parsing* are goal-directed and INCREMENTAL, and hence are governed by essentially the same mechanism. Under this model, a human hearer-parser builds a succession of partial parse trees based on what (s)he has heard thus far. Importantly, however, unlike the conventional bottom-up parsing, the DS model assumes a strong *predictive* element in parsing: a hearer is assumed to entertain some *goal* to be reached eventually at any stage of parsing. In (1), for example, as soon as the hearer encounters *Bob*, an underspecified *propositional* tree is constructed, as in the first simplified and schematised tree in Figure 1. Then the tree 'grows' *monotonically*, i.e. such that at each word input, it is 'updated' to an 'incremented' tree that is *subsumed* by the original tree, as depicted in the same Figure. This can be described as a process of specifying the relevant nodes towards a complete tree. This predictive element in DS allows a speaker-generator to be modelled as doing exactly the same, i.e. going through monotonically updated partial trees, the only difference being that (s)he also has available a more

---

[1]These *linked* structures are suppressed in all diagrams.

fully specified *goal tree* representing what (s)he wishes to say, corresponding to the rightmost tree in the Figure (with '0' in the 'generation' row at the bottom indicating it is entertained before utterance). Each licensed step in generation, i.e. the utterance of a word, is governed by whatever step is licensed by the parsing formalism, constrained via a required subsumption relation of the goal tree. By updating their growing "parse" tree relative to the goal tree, speakers are licensed to produce the associated natural language string.

Parsing: 1   2   3

$$?Ty(t) \qquad \mapsto \qquad ?Ty(t) \qquad \mapsto \qquad \begin{array}{c} Ty(t), \\ \textbf{Leave'(Bob')}, \Diamond \end{array}$$

$$\begin{array}{cc} Ty(e), & ?Ty(e \to t) \\ \textbf{Bob}', \Diamond & \end{array} \qquad \begin{array}{cc} Ty(e), & Ty(e \to t), \\ Bob' & \textbf{Leave}', \Diamond \end{array} \qquad \begin{array}{cc} Ty(e), & Ty(e \to t), \\ Bob' & Leave' \end{array}$$
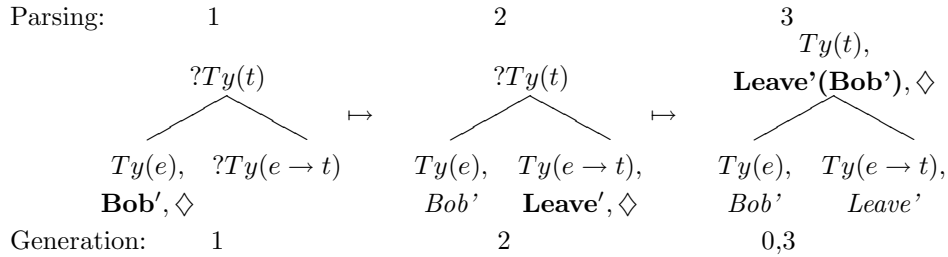
Generation: 1   2   0,3

Figure 1: Parallel parsing and generation in DS

This architecture allows for a dialogue model in which generation (what a speaker does) and parsing (what a hearer does) function in parallel. The speaker goes through partial trees subsuming a specified goal tree, while the hearer attempts to 'mirror' the same series of partial trees, albeit not knowing what the content of the unspecified nodes will be. For the dialogues in (1)-(3), therefore, B as the hearer will have the partial representation of what he has successfully parsed, required also for generation. This provides him with the ability at any stage to become the speaker, interrupting to ask for clarification, reformulating, or providing a correction, as and when necessary. As we shall see, B's parse tree reveals where need of clarification or miscommunication occurs, as it will be at that node from which a sub-routine extending it takes place[2]. According to our model of dialogue, repeating or extending a constituent of A's utterance by B is licensed only if B, the hearer of A turned now a speaker, entertains currently a goal tree that matches or extends the parse tree of what he has heard in a monotonic fashion, although he only utters the relevant subpart of A's utterance. Indeed, this update is what B is seeking to clarify, correct or acknowledge. In DS, B can reuse the already constructed (partial) parse tree in his context, rather than having to rebuild an entire propositional tree or subtree[3].

---

[2] The account extends the implementation reported in Purver et al. (2006)

[3] Given the DS concept of linked trees projecting propositional content, we anticipate that this mechanism will be extendable to fragment construal involving inference (see e.g. Schlangen (2003), Schlangen and Lascarides (2003))

4 NSU fragments in Dynamic Syntax

4.1 Non-repetitive Acknowledgement

From a DS perspective, phenomena like *reformulations* as in (1), or *extensions* to what one understands of the other speaker's utterance, (2), can be handled with exactly the same mechanisms as the sentence-internal phenomenon independently identifiable as *apposition*, as in (4), and equally usable by a single individual as a means of incrementally reformulating, correcting or extending what they have just uttered. The update rule for such structures, applicable to all terms, takes the two type e terms so formed and yields a new term whose compound content is a combination of both.

We now have the basis for analysing extensions potentially functioning as acknowledgements which build on what has been previously said as a way of confirming the previous utterance. Recall (1), (2). There are two ways for fragments which reformulate an interlocutor A's utterance to occur: (a) as interruptions of A's utterance with immediate confirmation of identification of the individual concerned, see (2); (b) as confirmations/extensions of A's utterance after the whole of her utterance has been integrated, see (1). Both are modelled by DS as incremental additions.

Turning to (1), B's response *(Yeah,) the accounts guy* constitutes a reformulation of A's utterance and an extension of A's referring expression, yielding a similar content as that of an appositive expression *Bob, the accounts guy* in this case jointly constructed. B's reformulation/extension counts in effect as an acknowledgement in virtue of signalling successful processing of A's utterance without objection raised. Thus there is no need for a separate grammatical mechanism to process these structures. In DS terms, after processing A's utterance, B's context consists of the following tree:

(7)   B's Context for producing 'Yeah'

$$Ty(t), Leave'(\iota, x, Bob'(x) \wedge Leave'(x)), \Diamond$$

$$(\iota, x, \overbrace{Bob'(x))} \qquad Leave'$$

B, as a speaker, can now re-use this representation as point of departure for generating the expression *the accounts guy*. In this case his goal tree, the message to be expressed, will now be annotated with a composite term made up from both the term recovered from parsing A's utterance and the new addition. This requires attaching a *linked* tree to the correct node and an appropriate update of the context tree (for reasons of space, the exact structure of the *linked* tree is condensed below, with subscripting as shorthand):

134

(8)   B's GOAL TREE:

$Ty(t), Leave'((\iota, x, Bob'(x) \wedge Acc.guy'(x))) \wedge Leave'(x))$

$(\iota, x, Bob'(x) \wedge Acc.guy'(x))$ $Leave'$ $(\iota, x, Bob'(x))_{Acc.guy'(x)}$

In order to license generation of the expression *the accounts guy*, B now needs to verify that processing these words in the context provided by the tree in (7) will produce a tree that matches this goal tree in (8). To achieve this, starting from (7), a series of simulated "parse" trees are generated which indeed result in the requisite matching. Steps include shifting the pointer to the appropriate node, projection of a *linked* tree from that node and test-processing the words *the accounts guy*, each step checking against the goal tree that a subsumption relation between the current "parse" tree and the goal tree is always maintained:

(9)   B's PARSE TREE licensing production of *the accounts guy*: LINK adjunction

$Ty(t), Leave'(\iota, x, Bob'(x) \wedge Leave'(x))$

$(\iota, x, Bob'(x))$ $Leave'$ $\mathbf{U}_{Acc.guy'(\mathbf{U})}, \Diamond$

The only way to update this representation relative to both the restriction on the metavariable and monotonicity of growth on any one node in a tree involves replacing the metavariable with $(\iota, x, Bob'(x))$, as this is commensurate with an extension of the term annotating the node from which the link transition was constructed:

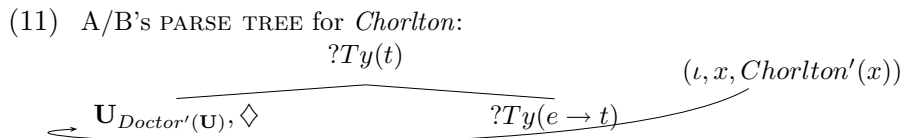(10)   Updating B's PARSE TREE licensing production of *the accounts guy*

$Ty(t), Leave'(\iota, x, Bob'(x) \wedge Acc.guy'(x) \wedge Leave'(x)), \Diamond$

$(\iota, x, Bob'(x) \wedge Acc.guy'(x))$ $Leave'$ $(\iota, x, Bob'(x))_{Acc.guy'(x)}$

Finally, the information is passed up to the top node of the main tree, completing the parse tree to match B's GOAL TREE, (8), thus licensing the utterance of the expression *the accounts guy*.

4.2   Non-repetitive Clarification

In the acknowledgement case above, the proposition relative to which the *linked* structure is built is completed (with an already extended epsilon term); but the same mechanism can be used when the interlocutor needs clarification, prior to any such completion of the tree. In (2), B again, as the speaker, takes as his goal tree a tree annotated with an expansion of the term constructed from parsing A's utterance but nevertheless picking out the same individual. Using the very same mechanism as in (1) of building a *linked* structure, B, interrupting A, provides a distinct expression, the name *Chorlton*, this time before he has completed the parse tree for A's utterance. All that has been achieved at this point is the definite's contribution of a meta-variable with the restriction that the individual picked out must be a doctor:

(11) A/B's PARSE TREE for *Chorlton*:

$$?Ty(t)$$

$$\mathbf{U}_{Doctor'(\mathbf{U})}, \Diamond \qquad\qquad ?Ty(e \to t) \qquad\qquad (\iota, x, Chorlton'(x))$$

As in the acknowledgement case, but this time at the node INITIATING the link transition, the only possible value to provide for the metavariable $\mathbf{U}$ compatible both with its restriction and the monotoniticity constraint is the composite term $(\iota, x, Doctor'(x) \wedge Chorlton'(x))$. The mechanism of constructing paired structures involving type e terms across *linked* trees is identical to that employed in B's utterance in (1), though to a rather different effect at this intermediate stage in the interpretation process. This extension of the term is confirmed by A, this time replicating the composite term which processing B's utterance has led to. The eventual effect of the process of inducing *linked* structures to be annotated by coreferential type e terms may thus vary across monologue and different dialogue applications, yielding different interpretations, but the mechanism is the same.
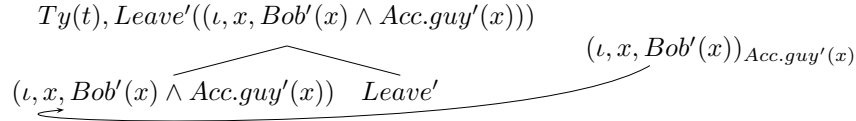
4.3   Correction

It might be argued nonetheless that correction is intrinsically a dialogue phenomenon. Consider (3), for example. One of the possible interpretations of (3), according to the DS analysis, is that B has offered the equivalent of the content derived by processing *Rob left?*. That is, let's assume here that B has misheard and requests confirmation of what he has perceived A as saying. A in turn rejects B's understanding of her utterance and provides more information. Presuming rejection as simple disagreement (i.e. the utterance has been understood, but judged as incorrect), in DS terms, this means that A has in mind a goal tree that licensed what she had produced, which is distinct from the one derived by processing B's clarification. As shown in Kempson et al. (2007), this means that A has been unable to process B's clarification request as an extension of her own context. Instead, she has to parse the clarification by exploiting the potential for introducing an initially structurally underspecified tree-node to accommodate the contribution of the word *Rob*. Subsequently, by re-running the actions stored in context previously by processing her own utterance of the word *left*, she is able to complete the integration of the fragment in a new propositional structure.

Now, in order for A to produce the following correction, what is required is for A to establish as the current most recent representation in context her ORIGINAL goal tree. This can be monotonically achieved by recovering and copying this original goal tree to serve as the current most immediate context[4]. An option available to A at this point is to introduce, in addition or exclusively, a reformulation of her original utterance in order to facilitate

---

[4]Mistaken representations must be maintained in the context as they can provide antecedents for subsequent anaphoric expressions.

identification of the named individual which proved problematic for B previously. She can answer B's utterance of *Rob?* with *(No,) Bob, the accounts guy*, as in (3) or simply with *(No,) the accounts guy*. Both are licensed by the DS parsing mechanism without more ado. For both, the goal tree will be as follows and it will always be the point of reference for checking the subsumption relation relative to the simulated parsing steps described further below:

(12) A's GOAL TREE

$$Ty(t), Leave'((\iota, x, Bob'(x) \wedge Acc.guy'(x)))$$

$$(\iota, x, Bob'(x))_{Acc.guy'(x)}$$

$$(\iota, x, Bob'(x) \wedge Acc.guy'(x)) \quad Leave'$$

Under these circumstances, given the DS grammar-as-parser perspective, several strategies are now available for the licensing of generation of the fragment. A is licensed to repeat the name *Bob* by locally extending the node in the context tree where the representation of the individual referred to is located by using the rule of LATE*ADJUNCTION, a process which involves building a node of type e from a dominating node of that type (illustrated in Kempson et al. 2007). An alternative way of licensing repetition of the word *Bob* is to employ one of the strategies generally available for the parsing of long distance dependencies i.e. constructing initial tree nodes as unfixed (*ADJUNCTION). We show here how the latter strategy can be exploited to license the production of the fragment by A.

(13) Parsing simulation licensing generation of *Bob, the accounts guy*

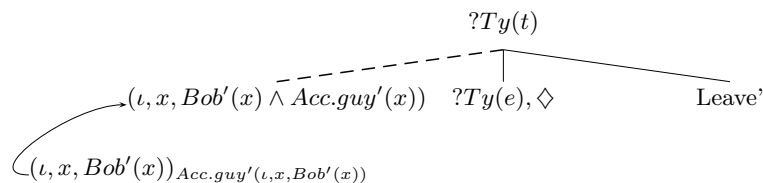Step 1: *Adjunction   Step 2: LINK-Adjuction + testing *the accounts guy*

$$?Ty(t)$$

$$?Ty(t)$$

$$(\iota, x, Bob'(x)), \diamondsuit$$

$$(\iota, x, Bob'(x)), \diamondsuit$$

$$\mathbf{U}_{Acc.guy'(\mathbf{U})}$$

The only way to develop the constructed tree at Step 2 commensurate with the goal tree (12) is to identify the value of $\mathbf{U}$ as $(\iota, x, Bob'(x))$, so this is what is entered at the newly constructed linked tree, duly leading to extension of the term originally given as annotating the unfixed node as $(\iota, x, Bob'(x) \wedge Acc.guy'(x))$. The structure[5] derived by processing such an extension is exactly that of (1) above (compare goal tree in (12) above and tree in (8)). Now, as mentioned before, *context*, as defined in DS, keeps track not only of tree representations and words but also of actions contributed by the words and utilised in building up the tree representations. Here, according to DS, production of the correction in (3) is licensed to be

---

[5]Again note that DS trees represent derived content rather than structure over natural language strings.

fragmental only because the original actions for parsing/producing the word *left* are available in the context and can be recalled to complete the structure initiated by processing/producing the name *Bob*. Now these stored actions can be retrieved to develop the tree further:

(14) Parsing simulation licensing generation of *Bob, the accounts guy*
Step 3: test-processing stored actions for *left*

$$?Ty(t)$$

$$(\iota, x, Bob'(x) \wedge Acc.guy'(x)) \quad ?Ty(e), \diamondsuit \qquad \text{Leave'}$$

$$(\iota, x, Bob'(x))_{Acc.guy'(\iota,x,Bob'(x))}$$

With this partial tree being commensurate with the goal tree, all actions that follow are general computational processes for completing the tree: unifying the unfixed node to determine the subject argument, applying the subject to the predicate, evaluating the quantified terms. Nothing specific to this structure is needed. Indeed, all these mechanisms are equally applicable by an individual speaker, perhaps more familiar as *right dislocation* phenomena, but equally available incrementally:

(15) Bob left, (Bob) the accounts guy.

5 Conclusion

As these fragments and their construal have demonstrated, despite serving distinct functions in dialogue, the mechanisms which make such diversity possible are general strategies for tree growth. In all cases, the advantage which use of fragments provides is a "least effort" means of re-employing previous content/structure/actions which constitute the minimal local *context*. As modelled in DS, it is more economical to reuse information from this local context rather than constructing representations afresh (via costly processes of lexical retrieval, choice of alternative parsing strategies, etc.).

A further quandary in dialogue construal is that, despite such avenues for economising their efforts, interlocutors are nevertheless faced with an increasing set of interpretive options at any point during the construction of representations. One strategy available to hearers is to delay a disambiguating move until further input potentially resolves the uncertainty. However, as further input is processed and parsing/interpretive options increase rapidly, the human processor struggles. The incremental definition of the DS formalism allows for the modelling of an alternative strategy available to hearers: at any (sub-sentential) point they could opt to intervene immediately, and make a direct appeal to the speaker for more information as illustrated by the clause-medial fragment interruption (2). It seems clear that the grammar should allow the resources for modelling this behaviour without any complications.

The phenomena examined here are also cases where speakers' and hearers' representations, despite attempts at coordination, may nevertheless separate sufficiently for them to have to seek "repair" (see especially (3)). In the model presented here, the dynamics of interaction allow fully incremental generation and integration of fragmental utterances so that interlocutors can be taken to constantly provide optimal evidence of each other's representations with necessary adjuncts being able to be incrementally introduced. But such mechanisms apply equally within an individual utterance, with *self-correction*, *extension*, *elaboration*, *repetition* etc. The effect is that all the devices which seem so characteristic of dialogue involve mechanisms invariably available within an individual's core grammar. This suggests a new inverse methodology: it is the challenge of modelling dialogue that can be used as a point of departure for modelling grammars for individual speakers, rather than the other, more familiar, way round (see also Ginzburg (forthcmg)). This reversibility is, notably, straightforwardly available to grammar formalisms in which the incremental dynamics of information growth is the core structural concept because emergent dialogue structure crucially exhibits and interpretively relies on such incrementality.

Acknowledgements

References

Patrick Blackburn and Wilfried Meyer-Viol. Linguistics, logic and finite trees. *Bulletin of the IGPL*, 2:3–31, 1994.

Raquel Fernández. *Non-Sentential Utterances in Dialogue: Classification, Resolution and Use.* PhD thesis, King's College London, University of London, 2006.

Jonathan Ginzburg. *Semantics for Conversation.* CSLI, forthcmg.

Ruth Kempson, Andrew Gargett, and Eleni Gregoromichelaki. Clarification requests: An incremental account. In *Proceedings of the 11th Workshop on the Semantics and Pragmatics of Dialogue (DECALOG)*, 2007.

Wilfried Meyer-Viol. *Instantial Logic.* PhD thesis, University of Utrecht, 1995.

Matthew Purver, Ronnie Cann, and Ruth Kempson. Grammars as parsers: Meeting the dialogue challenge. *Research on Language and Computation*, 4(2-3):289–326, 2006.

David Schlangen. *A Coherence-Based Approach to the Interpretation of Non-Sentential Utterances in Dialogue.* PhD thesis, University of Edinburgh, 2003.

David Schlangen and Alex Lascarides. The interpretation of non-sentential utterances in dialogue. In *Proceedings of the 4th SIGdial Workshop on Discourse and Dialogue*, pages 62–71, Sapporo, Japan, July 2003. Association for Computational Linguistics.

Noor van Leusen and Reinhard Muskens. Construction by description in discourse representation. In J. Peregrin, editor, *Meaning: The Dynamic Turn*, chapter 12, pages 33–65. 2003.