Grammars as models for language use: implications and consequences
Extended abstract for IPrA panel
“The Official and the Unofficial Business of Conversation”
(2011, Kerstin Fischer, Gregory Mills)

Eleni Gregoromichelaki  Ruth Kempson
eleni.gregor@kcl.ac.uk  ruth.kempson@kcl.ac.uk
King’s College London  King’s College London

with
Matthew Purver  Ronnie Cann
mpurver@dcs.qmul.ac.uk  r.cann@ed.ac.uk
Queen Mary University of London  University of Edinburgh

July 1, 2011

1 Introduction

Following Chomsky (1965), there has been broad agreement, until recently, that formal accounts of natural language (NL) grammars must be grounded in the description of sentence-strings without any reflection of the dynamics of language performance. Departures from this anti-functionalism were rejected on the basis that language use is often disfluent and disorderly, hence presumed to preclude rigorous systematization, a stance independently propounded by the anti-formalist approach of Ordinary Language philosophy (Austin 1975) and followed up by many theoretical approaches to pragmatics. However, amenability to formalization and reflections of performance are far from incompatible (Newmeyer 1998), as witness the huge growth in context-modelling and information update in formal semantics since the development of DRT and related frameworks. Such de-
velopments are now beginning to show that the methodology of standard grammar formalisms is problematic in that all phenomena of NL context-dependency are explainable only by bifurcating them into grammar-internal vs. grammar-external/discourse processes. This is because NL grammars are, on the one hand, taken to be limited to phenomena occurring within sentence boundaries but, on the other, unable to reflect the incremental word-by-word comprehension and production at the sub-sentential domain. However, context-dependent phenomena - anaphora, ellipsis, tense-construal, quantification, etc. - all allow unified resolutions of how they are to be understood within and across sentence boundaries and even across distinct interlocutor turns in dialogue (Purver et al 2010; Gregoromichelaki et al 2010).

Given this standard view of grammar, as independent of language use, even (psycholinguistic) models within the language-as-action tradition bifurcate the concept of ‘language’ as language_u (that is, language-in-use), to be distinguished from language_s (that is, language structure). For example, Clark (1996) sees language just as a means of coordination in joint activities with some of its context-dependency features as derivative from its function as a coordination device:

Many phenomena have been treated as phenomena of language use when they are really features of the joint activities in which language is being used. (Clark 1996: 388)

In this paper we suggest that these bifurcations - language use vs. language structure, competence vs. performance, grammatical vs. psycholinguistic/pragmatic modes of explanation are all based on an arbitrary and ultimately mistaken dichotomy of phenomena, one that obscures their unitary nature because it insists on a view of grammar that ignores essential features of NL processing.

In a similar vein, a number of researchers have recently pointed out that a range of metacommunicative acts (in track 2: Clark 1996) running in parallel with the communicative acts (in track 1) have to be characterised as part of the grammar itself (e.g. Purver 2006; Fernandez 2006; Ginzburg forthcoming; Gregoromichelaki et al forthcoming). As a response to such considerations, grammatical models have recently begun to appear that reflect aspects of performance to varying degrees (Hawkins 2004; Phillips 1996; Lombardo and Sturt 2002; Ginzburg and Cooper 2004; Kempson et al. 2001, Cann et al. 2005). One such model, Dynamic Syntax (DS), has the distinctive characteristic of taking a fundamental feature of real-time processing - the concept of underspecification and incremental goal-directed update - as the basis for grammar formulation. This simple shift of perspective has enabled the modelling of core syntactic phenomena as well as phenomena at the syntax-semantics-pragmatics interface in a unified and hence explanatory way. At the heart of this approach is the assumption that grammatical constraints are all defined in terms of the progressive growth of representations of content, with
partial interpretations built more or less on a word-by-word basis (see also Sturt and Crocker 1996). Language production is also argued to be incremental (Levelt 1989; Ferreira 1996) and based on exactly the same mechanisms, hence allowing the interleaving of planning, syntactic structuring of the message and articulation, reflecting the introspective observation that the end of a sentence is not planned when one starts to utter its beginning.

In accordance with these assumptions, in dialogue, evidence for radical incremen
tality is provided, not merely by the fact that participants incrementally “ground” each other’s contribution (Allen et al. 2001) through back-channel contribu
tions like yeah, mhm, etc., but also by the fact that people clarify, repair and extend each other’s utterances, even in the middle of an emergent clause —split utterances:

(1) Context: Friends of the Earth club meeting
A: So what is that? Is that er... booklet or something?
B: It’s a book
C: Book
B: Just ... talking about al you know alternative
D: On erm... renewable yeah
B: energy really I think......
A: Yeah [BNC:D97]

In fact, such completions and continuations have been viewed by Herb Clark, among others, as some of the best evidence for cooperative behaviour in dialogue (Clark, 1996, 238). Instead of ignoring such data as beyond the remit of grammars, DS takes the view that joint-construal of meaning in dialogue is fundamentally based on the same mechanisms underlying language structure: structure is built through incremental procedures, that integrate context in every step, and this pro
vides principled explanations for the structural properties of linguistic signals; in addition, since the grammar licenses partial, incrementally constructed structures, speakers can start an utterance without a fully formed intention/plan as to how it will develop relying on feedback from the hearer to shape their utterance (Goodwin 1979) and its construal –this provides the basis for the joint derivation of structures, meaning and action in dialogue.

1.1 Split Utterances

Split utterances (SUs) – utterances split across more than one speaker or dialogue contribution – are common in spontaneous conversation and provide an important source of data that can be used to test the adequacy of linguistic theories (Purver
et al. (2009b)). Previous work has suggested that Dynamic Syntax (DS) is well placed to analyse these phenomena as it is strictly word-by-word incremental, allowing an account of speaker changes at any point (Purver et al. (2006)) and interruptive phenomena such as mid-sentence clarification sequences (Gargett et al. (2009)). However, other less incremental grammar formalisms have also been applied to particular kinds of SUs: Poesio and Rieser (2010) use LTAG\footnote{Unlike DS, LTAG must be supplemented by a parsing and/or generation model (a set of defeasible inference rules for Poesio and Rieser) to derive the incrementality required.} in an analysis of collaborative completions. But given that such accounts employ grammars which license strings of words a direct account for SUs is prevented as the reference and binding of indexical speaker/addressee pronouns changes as the speaker transition occurs:

(2) A: Did you give me back
   B: your penknife? It’s on the table.

(3) A: I heard a shout. Did you
   B: Burn myself? No, luckily.

A grammar must rule out the sentence did you burn myself? as ungrammatical if spoken by one single speaker, but allow it as grammatical if the identity of the speaker changes as in (3) – this will be hard for a string-based account. In (2), you and your must be able to take different referents due to the speaker change between them. In contrast, as DS defines grammatical constraints in terms of the incremental construction of semantic content (rather than through licensing strings via an independent layer of syntax over strings), we show that such examples are not problematic given an independently motivated definition of the lexical entries for indexicals.

SUs can also perform diverse dialogue functions, with the speech acts associated with the individual speakers’ contributions often being different – see Purver et al. (2009b). In (2)-(3), B’s continuations seem to function as clarifications of A’s intended queries. Others have pointed out that continuations can function as e.g. adjuncts (Fernández and Ginzburg (2002)) or clarification requests (Purver et al. (2003)); and Poesio and Rieser (2010) show how a completion (in their terms) can get its function in the dialogue (in their case, to act as a collaborative contribution to a plan). A full account of SUs therefore requires some representation of such dialogue function information in the model of context that guides dialogue interpretation and production.

To account for such phenomena, the processing model needs to incorporate the means for deriving appropriate indications of illocutionary force or dialogue act type, even when derivation of such information is not linguistically determined. In
the case of SUs, the linguistic input on its own provides no such indications; instead, the grammar provides adequate means of continuing/taking over somebody else’s utterance, and this does not necessarily involve strategic reflection or fully-formed intentions as to what function the utterance should perform: this provides the possibility for speakers to ‘blurt out’ utterances without necessarily having any specific plans/intentions in mind, and for hearers so respond without reflection as to the speaker’s plan. But, as pointed out in Kempson et al. (2007), Gargett et al. (2009), DS provides mechanisms for allowing the inclusion of optional inferred information. We present here an extension to DS which allows it to include such information explicitly and draw such distinctions relevant for SUs. We also show how this extension is motivated by the resolution of the Split Turn Taking Puzzle (STTP). This is a version of Ginzburg (1997)’s Turn Taking Puzzle applied to SUs, where it appears that distinct empirical results are obtained: given an SU split between two people, the possible interpretations of a subsequent “Why?” depend not on the most recent speaker, but on who can be taken as the agent of the speech act performed – which may be distinct from the notion of ‘speaker’ tracked by indexical pronouns like *I* and *my*.

The view that emerges, instead of relying on mind-reading and metarepresentational mechanisms, entails a reconsideration of ‘signalling’ (or ‘ostensive communication’) in a naturalistic direction and a non-individualistic view on ‘meaning’ (Millikan 1993, 2005). Coordination/alignment/inter-subjectivity among dialogue participants relies on low-level mechanisms (Pickering and Garrod 2004; Mills and Gregoromichelaki 2010, Böckler et al 2010, Gregoromichelaki et al 2011) like the grammar (appropriately conceived).

2 Dynamic Syntax

In response to the challenge that SU data provide, we now turn to a particular grammar formalism, Dynamic Syntax, to consider whether forms of correlation between parsing and generation, as they take place in dialogue, can provide a basis for modelling recovery of interpretation in communicative exchanges. We set out a model of parsing and production mechanisms that makes it possible to show how, with speaker and hearer using incrementally the same mechanisms for construal, issues about interpretation choice and production decisions may be resolvable solely on the basis of feedback, without reflections on the other party’s mental state. As we shall see, according to this account (Purver et al 2006), what underpins the smooth shift in all joint endeavours of conversation is the incremental, context-dependent processing shared by parsing and generation, and the tight coordination thereby achievable (similar assumptions underpin the model presented in Stone
2004, 2005, even though distinct conclusions are drawn there as to its implications with respect to the issue of intention recognition in communication).

Instead of data such as (1)-(3) being problematic, extensive use of mechanisms across interlocutors illustrates the advantages of a DS-style incremental, dynamic account over static models. The incremental licensing of word processing modelled by DS, directly provides for the construction of restricted, contextually salient structural frames within which fragment construal/generation takes place. From a parsing perspective, this allows narrowing down of the threatening multiplicity of interpretations by incrementally weeding out possibilities en route to some commonly shared understanding. But the features of incrementality, predictivity/goal-directedness and context-dependent processing are built into the grammar architecture itself, rather than being external factors imposed by parsing/production mechanisms: each successive processing step relies on a grammatical apparatus which integrates lexical input with essential reference to the context in order to proceed. Under this low-level licensing of incrementally expanding strings and their interpretations, no mechanisms trigger high-level decisions about speaker/hearer intentions as part of the grammar itself. Rather, participants are modelled as gradually shaping propositional contents, on a word-by-word basis, drawing on sub-personal, synchronised mechanisms, without having to start with a fully-formed truth-evaluable content in mind. Such a view is buttressed by the fact that, as (4)-(6) show (repeated below), neither party in such role-exchanges are able to know the eventual joint proposition in advance:

(4) Daughter: Oh here dad, a good way to get those corners out
    Dad: is to stick yer finger inside.
    Daughter: well, that’s one way.

(5) (A and B arguing:)
    A: In fact what this shows is
    B: that you are an idiot

(6) (A mother, B son)
    A: This afternoon first you’ll do your homework, then wash the dishes and then
    B: you’ll give me £10?

Our DS-based claim then is that communication involves taking risks without requiring mind-reading as an essential attribute: success in communication thus characteristically involves cycles of clarification/correction/extension/reformulation...
etc ("repair strategies", in Track 2: Clark 1996) as essential subparts of the exchange. When modelled non-incrementally, such strategies might lead to the impression of non-monotonic repair operating at a distinct representational level, and the need to revise some otherwise stable context. But pursued incrementally, within a goal-directed architecture, as we shall see, these do not constitute communication breakdown or disfluencies, but the normal mechanism of context construction, hypothesised update, and confirmation (see also Schegloff 1979). By building on the assumption that successful communication may crucially involve subtasks of repair (Ginzburg forthcoming), mechanisms for informational update that underpin interaction can be defined without reliance on (meta-)representing contents of the interlocutors’ mental states as a precondition for successful communication.

2.1 Dynamic Syntax: the formalism

DS is a procedure-oriented framework modelling sequential processing. Parsing in DS relies on the execution of licensed actions, as incorporated in lexical entries (as in (7) below); such actions resolve outstanding requirements (here, $?Ty(e)$) to decorate the tree with information about semantic type $Ty$ and content (formula) $Fo$:

$$\text{John:} \begin{cases} \text{IF} & ?Ty(e) \\ \text{THEN} & \text{put}(Ty(e)) \\ & \text{put}(Fo(John')) \\ \text{ELSE} & \text{abort} \end{cases}$$  

(7)

"John arrived" $\mapsto \Box, ?Ty(t)$  

(8)

Application of lexical actions is interspersed with the execution of computational rules which provide the predictive element in the parse and provide the compositional combinatorics. For example, eventual type deduction and function application is achieved by means of the rule of Elimination (9). This derives the value of a mother node’s semantic type $Ty$ and content $Fo$ from that of its daughters, in (8) providing the values $Ty(t), Fo(Arrived'(John'))$ at the top node:
Elimination:
IF \( ?Ty(T_1), \)
\[ \downarrow_0 (Ty(T_2), Fo(\alpha)) \]
\[ \downarrow_1 (Ty(T_2 \rightarrow T_1), Fo(\beta)) \]
THEN put\( (Ty(T_1)) \)
put\( (Fo(\beta(\alpha))) \)
ELSE abort

“Completion, Elimination”
\[ \diamondsuit, ?Ty(t), Fo(Arrived'(John')) \]

Grammaticality is then defined in terms of a resulting complete (requirement-free) tree. As is displayed in (12) by way of illustration, the build up of interpretation for (11) is monotonic and strictly word-by-word incremental:
As (12) illustrates, the DS system provides mechanisms that enable the hearer to anticipate the structure of the incoming signal and therefore it allow incremental word-by-word build-up of representations of content paired with some word string. Amongst such predictive steps are the construction by anticipation of a subject-predicate schema (stages 0-1 above, with requirements for a subject and predicate
imposed as a very first step (not illustrated here), and their immediate construction at the second). Such a frame then makes possible the identification of the subject as some individual named John, via processing of the word John (stage 2), and then successive steps of identifying the predicate and its internal argument to be paired with verb and object noun-phrase respectively (stages 3-4). These updates then provide input to the compilation by labelled type-deduction of a propositional representation of content (stage 4). This then as a final step is subject to an algorithm of evaluation determining how some assigned scope dependency choices are reflected in the constructed names (here the formula $S < x$ indicates that the existential term binding the variable $x$ is taken as dependent on the event-term $S$ (see Gregoromichelaki 2011, Cann forthcoming). The mechanisms for tree growth and evaluation are identically available to speakers, hence in generation. The only essential difference in production is that the modelling of a speaker’s actions for tree-growth update involve a so-called “goal tree” (tree 4 in (12)) relative to which all intermediate construction steps have to be checked for commensurability, a checking step for which there may be no analogue in parsing (see section 3.2).

The notion of incrementality in DS is closely related to another of its features, the goal-directedness/predictivity of both parsing and generation (Demberg-Winterfors 2010, see also). At each stage of processing, structural predictions are triggered that could fulfill the goals compatible with the input, in an underspecified manner. Representations of the conceptual structure of messages are given as binary trees, formally encoded with the tree logic $LOFT$ Blackburn and Meyer-Viol (1994). $LOFT$ is a modal logic with operators $⟨↑⟩, ⟨↓⟩, ⟨↑∗⟩, ⟨↓∗⟩$ to define the relations of immediate and iterative domination, and to indicate node locations. What is novel about such trees is, on the one hand, that though they constitute a form of syntax, they are not inhabited by words of the language – they constitute structures inhabited by (lambda binding) formulae in the epsilon calculus, the selected semantic representation language. Furthermore, the mechanisms that define such progressive tree construction constitute the sole concept of natural-language syntax which the DS grammar provides. The system is goal-directed; and trees are constructed, by starting (in the context-independent case) from a radically underspecified goal, the axiom (the leftmost minimal tree in the illustration provided by (12), and proceeding through monotonic updates of partial or structurally underspecified trees until some tree is constructed from an input string in which all imposed goals and subgoals are met. Every node in a complete tree bears annotations that include the semantic formulae and their type information.

Crucial for expressing the goal-directedness are requirements, i.e. unrealized but expected node/tree specifications, indicated by ‘?’ in front of annotations. As the axiom and its immediate subsequent update tree development in (12) indicate,
requirements may also take a modal form, e.g. the constraint $\langle \downarrow \rangle Ty(e \rightarrow t)$, which is a constraint that the daughter be a formula of predicate type. Requirements are essential to the tree-growth dynamics. All requirements must be satisfied if the construction process is to lead to a successful outcome, and, as indicated by the requirement for the predicate imposed at stage 2 in (12), these may not be satisfied until substantially later than the point at which they are imposed.$^2$

Updates are carried out by means of applying both computational and lexical actions, which introduce and update nodes, and move the pointer. Computational actions govern general tree-constructional processes in a broadly top-down manner.$^3$ Lexical specifications, equally, induce actions that effect tree-development, providing annotations for nodes, in many cases also inducing the construction of further structure.$^4$ In the update from stage 2 to 3 in (12), for example, the set of lexical actions for the word see is applied, yielding the predicate subtree and its annotations. Subsequent computational actions involve progressive labelled type-deduction decorating non-terminal nodes in the tree strictly bottom-up until the goal defined in the axiom is reached. Indeed all actions, computational and lexical, are defined in the same tree-growth vocabulary, so there is free intercalation of the various types of process. Thus partial trees grow incrementally, driven by procedures associated with particular words as they are encountered while conforming to top-down modal requirements on later development.

Central to the framework is the modelling of quantification as a process of term construction, using the epsilon calculus as the basic formula language (the epsilon calculus is the formal language that employs arbitrary-name terms in predicate logic natural-deduction proofs). All terms are of type $e$: epsilon terms, as illustrated by $(e, x, \text{Consultant}'(x))$. This term constitutes an arbitrary witness of the existentially quantified formula $\exists x.\text{Consultant}'(x)$, as defined by the following equivalence:

$$\psi(e, x, \psi(x)) \equiv \exists x.\psi(x)$$

Notice how this equivalence yields the effect that an epsilon term invariably reflects its containing environment (the predicate $\psi$ in the term’s restrictor is a duplicate of the predicate applying on the term). The construction of such terms is induced by actions which incrementally, in part lexically, specify and collect up scope con-

---

$^2$The pointer, $\diamond$, indicates the ‘current node’ in processing, namely the one to be processed next, a constraint which governs word order.

$^3$This is the characterisation of incrementality adopted by some psycholinguists under the appellation of connectedness (Sturt and Crocker (1996)): an encountered word always gets ‘connected’ to a larger, predicted, tree.

$^4$For cases of dislocation, DS employs unfixed nodes (not developed in this paper) which is indeed a core notion: such nodes are initially assigned structurally underspecified positions that are subsequently updated (at the point of the gap, in transformational parlance): see Kempson et al 2001, Cann et al 2005 among others.
straints of the form \( x < y \) (to be understood as the term with variable \( y \) is dependent on the term with variable \( x \)). For example, indefinites project epsilon terms subject to the constraint that they are invariably dependent on either another quantifying expression or a term within the temporal specification; names, as iota terms (e.g. \( \iota, x, John' (x) \)), are, in contrast, taken to be epsilon terms of widest scope. A final algorithmic step yields the complex structure of the resulting terms as required in the equivalence: thus \( A \) consultant arrived is assigned a propositional formula

\[
Arrived'(\epsilon, x, Consultant'(x))
\]

is evaluated as

\[
Consultant'(a) \land Arrived'(a)
\]

where

\[
a = (\epsilon, x, Consultant'(x) \land Arrived'(x))
\]

The overall dynamics is thus one of growth in names as well as in structure.

More radical underspecification of formulae at intermediate stages, equally associated with a process of growth, is lexically licensed, for example by pronouns, which act as simple place-holders for some possibly subsequent identification. These are defined as projecting a metavariable (notated as \( U, V \) etc) as a place-holder for some value to be assigned, with an associated type specification, for pronouns \( Ty(\epsilon) \). These invariably occur with an associated requirement for a fixed formula value (of the form \( ?\exists xFo(x) \)), making such provision of a value essential to a successful outcome. Metavariables are substituted by other terms available in the context as part of the construction process, subject to locality restrictions differentiating e.g. pronouns and reflexives (for details see Cann et al. 2005, Kempson et al. 2001b). A distinctive DS flavour lies in the license for a parse to proceed on the basis of such partial information. Indeed, given the type specification but lack of formula value in the processing of a pronoun, the value for such metavariables may be able to established somewhat later, as for example in expletive uses of pronouns, as in \( \text{It is likely that Geoff is wrong} \).

In addition to the construction of individual predicate-argument structures, complex trees are obtained through a general tree-adjunction operation that licenses the construction of so-called linked trees. These are pairs of trees sharing information in the form of a shared term, each such tree a subdomain in which labelled type-deduction takes place as in the simple structures. These provide a grammar-induced structural form of context. The construction processes determining and then updating such partial tree representations are used to model a

---

5 We simplify presentation here by indicating the content of e.g. a name like \( John \) as \( John' \).
For example, in taking definite NPs to be anaphoric, we define the definite article as introducing a metavariable as a partial term, inducing also construction of a LINK transition to allow the construction of a tree providing possibly complex information as a constraint (“presupposition”) on the value to be substituted for the constructed partial term:

(13) The man smokes.

```
(14) \[ \text{the man} \mapsto ?Ty(t) \]
   \[ Ty(e), \ U, \triangle \mapsto Ty(e \rightarrow t) \]
   \[ Ty(t), \text{Man}'(\U) \]
   \[ Ty(e), \ U \]
   \[ Ty(e \rightarrow t), \text{Man}' \]
```

The structure on the subject node above is abbreviated as: \( Ty(\epsilon), \ U_{\text{Man}'(\U)} \).

Appositional structures, as in a consultant, a friend of Jo’s, left, can equally be established as inducing a pair of LINKed structures. A LINK transition is defined with the effect shown in (15) from a node of type \( \epsilon \) in which a preliminary epsilon term has been constructed onto a LINKed tree introduced with a requirement to develop a term using that very same variable:

(15) a friend of Jo’s

```
(15) \[ \text{a friend of Jo’s} \mapsto ?Ty(t) \]
   \[ Ty(e), (\epsilon, x, \text{Consultant}'(x)) \wedge \text{Friend}'(Jo')(x) \mapsto Ty(e \rightarrow t) \]
   \[ Ty(cn), (x, \text{Consultant}'(x)) \]
   \[ Ty(cn \rightarrow e), \lambda P.\epsilon, P \]
   \[ Ty(cn), (x, \text{Friend}'(Jo')(x)) \]
   \[ Ty(cn \rightarrow e), \lambda P.\epsilon, P \]
   \[ x \]
   \[ \text{Friend}'(Jo') \]
   \[ \text{Friend}' \]
```

---

6 The canonical case is relative clause construal (Kempson et al. 2001b, Cann et al. 2005), where some type \( \epsilon \) term once processed becomes the context for the projection of one such LINKed structure, which, when completed, allows the pointer to return to that initial type \( \epsilon \) term, now enriched by the incorporation of information constructed upon such a LINKed (adjunct) tree.

7 In (15), we abbreviate the annotation \((\iota, y, Jo'(y))\) to \( Jo' \) for simplicity.
A twinned evaluation rule then combines the restrictors of two such paired terms to yield a composite term on the main tree (unlike the P&R account, this does not involve ambiguity of the head NP according to whether a second or subsequent NP follows). The fact that the first term has not been completed is no more than the term-analogue of the delaying tactic made available by expletive pronouns and extraposition-from-NP constructions, whereby a parse can proceed from some type specification of a node (with attendant metavariable as its formula value), but without completing (evaluating) that formula. Just as with expletives, this strategy allows term modification when the pointer returns from its sister node to that only partially constructed term immediately prior to compiling the decorations of its mother:

(16) A man has won, someone you know.

Such linked trees and their development set the scene for a general characterisation of context, ranging over possibly partial trees and their updates. Context in DS is defined as the storage of parse states, i.e., the storing of partial tree, word sequence parsed to date, plus the actions used in building up the partial tree. Formally, a parse state $\mathcal{P}$ is defined as a set of triples $\langle T, W, A \rangle$, where: $T$ is a (possibly partial) tree; $W$ is the associated sequence of words; $A$ is the associated sequence of lexical and computational actions (Cann et al. 2007). At any point in the parsing process, the context $\mathcal{C}$ for a particular partial tree $T$ in the set $\mathcal{P}$ can be taken to consist of: a set of triples $\mathcal{P}' = \{\ldots, \langle T_i, W_i, A_i \rangle, \ldots\}$ resulting from the previous sentence(s); and the triple $\langle T, W, A \rangle$ itself, the subtree currently being processed. Anaphora and ellipsis construal generally involve re-use of formulae, structures, and actions from the set $\mathcal{C}$. All fragments illustrated above in (1)-(18) are processed by means of either extending the current tree, or by constructing linked structures with transfer of information among them so that one tree provides the context for another. Such fragments are licensed as wellformed by the grammar only relative to such contexts (Cann et al. 2007, Gargett et al. 2008, Kempson et al. 2009).

2.2 Parsing/generation coordination

This architecture allows a dialogue model in which generation and parsing function in parallel, following exactly the same procedure in the same order. Returning to (12), we now pick out the generation steps involved in producing Bob saw Mary, notated as (compressed) stages 0 to 4. As indicated earlier, generation of this utterance follows precisely the same actions and trees from left to right as in parsing, with the one additional filter, that the complete tree is available as a goal tree from
the start (hence the labelling of the complete tree as $T_g$). The intuition this reflects is that the eventual message, in this simple context-independent case at least, is known in advance by the speaker and determines the choices to be made. What generation involves, in addition to parse steps, is reference to $T_g$ to check whether each attempted generation stage (1, 2, 3, 4) is consistent with it. According to this algorithm, a subsumption check is carried out as to whether the current parse tree is monotonically extendible to $T_g$. The trees 1-3 are licensed because, for each of these, the subsumption relation to $T_g$ is maintained. Each time then the generator applies a lexical action, it is licensed to produce the word that carries that action only under successful subsumption check: at stage 3, for example, the generator processes the lexical action which results in the annotation $\text{See}^\prime$, and upon success and subsumption of $T_g$ license to generate the word $\text{see}$ ensues.

For processing split utterances, two more consequences are pertinent. First, there is nothing to prevent speakers initially having only a partial structure to convey, i.e. $T_g$ may be a partial tree: this is unproblematic, as all that is required by the formalism is monotonicity of tree growth, and the subsumption check is equally well defined over partial trees. Second, the goal tree $T_g$ may change during generation of an utterance, as long as this change involves monotonic extension; and continuations/reformulations/extensions across speakers are straightforwardly modelled in DS by appending a linked structure annotated with added material to be conveyed (preserving monotonicity) as in single speaker utterances:

(17) A friend is arriving, with my brother, maybe with a new partner.

Such a model under which the speaker and hearer essentially follow the same sets of actions, each incrementally updating their semantic representations, allows the hearer to mirror the same series of partial trees as the producer, albeit not knowing in advance the content of the unspecified nodes. Furthermore, not only can the same sets of actions be used for both processes, but also a large part of the parsing and generation algorithms is shared. In particular, the processing actions of both parsing and production involve the same progressive growth of partial tree representations, this being the only concept of “syntax” in the DS model. Even the concept of goal tree, $T_g$, may be shared between speaker and hearer, in so far as the hearer may have richer expectations relative to which the speaker’s input is processed, as in the processing of a clarification question. Conversely, the speaker may have only a partial tree as $T_g$, relative to which they are seeking clarification.

In general, as no intervening level of syntactic structure over the string is ever computed, the parsing/generation tasks are more parsimonious in terms of repre-

---

\footnote{In fact, the goal tree for the speaker need only be subsumed by at least one parse step, and in all nonfinal steps in the generation process non-trivially.}
sentations than in other frameworks. Additionally, the top-down architecture in combination with partiality allows the framework to be (strategically) more radically incremental in terms of interleaving planning and production than is possible within other frameworks. On the one hand, there is one less level of representation to be computed, so no need for a complex step-by-step correlation of syntactic and semantic output, and no recourse either to some externally imposed parser to ensure such correlation. On the other hand, the licensing of partial structures allows articulation before a complete propositional goal has been determined and, therefore, interlocutor suggestions can be integrated without the need for revision.

3 Split utterances in Dynamic Syntax

Split utterances follow as an immediate consequence of these assumptions. For dialogues (1)-(3) and (4)-(6), A reaches a partial tree of what she has uttered through successive updates, while B as the hearer, follows the same updates to reach the same representation of what he has heard: they both apply the same tree-construction mechanism which is none other than their effectively shared grammar. This provides B with the ability at any stage to become the speaker, interrupting to continue A’s utterance, repair, ask for clarification, reformulate, or provide a correction, as and when necessary. According to DS assumptions, repeating or extending a constituent of A’s utterance by B is licensed only if B, the hearer now turned speaker, entertains a message to be conveyed (a new $T_g$) that matches or extends in a monotonic fashion the parse tree of what he has heard. This message (tree) may of course be partial, as in (18), where B is adding a clarificational linked structure to a still-partially parsed antecedent, or it may complete the tree as in (19) and elsewhere:

(18) A: They X-rayed me, and took a urine sample, took a blood sample. Er, the doctor
B: Chorlton?
A: Chorlton, mhmm, he examined me, erm, he, he said now they were on about a slide (unclear) on my heart. [BNC: KPY 1005-1008]

(19) with smoke coming from the kitchen:
A: I’m afraid I burnt the kitchen ceiling
B: But have you
A: burned myself? Fortunately not.

9 A completely identical grammar is, of course, an idealisation but one that is harmless for current purposes.
Importantly, in DS, both A and B can now re-use the already constructed (partial) parse tree in their immediate context as a point from which to begin parsing and generation, rather than having to rebuild an entirely novel tree or subtree. By way of illustration, we take a simplified variant of (19):

(20) Ann: Did you burn

Here, of course, the reconstruction of the string as *Did you burn myself? is unacceptable (at least with a reflexive reading of myself), illustrating the problem of purely syntactic accounts of split utterances. But under DS assumptions, with representations only of informational content, not of putative structure over strings of words, the switch of person is entirely straightforward. Consider the partial tree induced by parsing A’s utterance Did you burn which involves a substitution of the metavariable projected by you with the name of the interlocutor/parser:

\[ \text{Did you burn} \mapsto ?Ty(t), Q \]
\[ ?Ty(e), Ty(e), Ty(e \rightarrow t) \]
\[ ?Ty(e), Ty(e \rightarrow (e \rightarrow t)), Burn' \]

At this point, Bob can complete the utterance with the reflexive as what such an expression does, by definition, is copy a formula from a local co-argument node onto the current node, just in case that formula satisfies the conditions set by the person and number of the uttered reflexive, in this case, that it names the current speaker:

\[ \text{myself} \mapsto ?Ty(t), Q \]
\[ Ty(e), Ty(e \rightarrow t) \]
\[ Ty(e \rightarrow (e \rightarrow t)), Burn' \]

Hence the absence of a “syntactic” level of representation distinct from that of such semantic representations allows the direct successful integration of such fragments.

---

10The feature $Q$ on a decorated node is not taken to have a fixed speech-act content: given the range of acts achievable by interrogative structures (as diverse as yes-no questions, wh questions, tag questions, exclamatives, etc.) we take interrogative forms to encode a direction by the speaker to the hearer for a particular type of coordination, here notated simply as $Q$. 

17
through the grammatical mechanisms themselves, rather than necessitating their analysis as sentential ellipsis.

Further, to illustrate how DS can sidestep the problems posed by abstraction accounts of ellipsis, we take a simplified version of (18):

(23)  A: The doctor

           B: Chorlton?

After processing *the doctor* both A and B share a context comprising a partial tree as follows:

(24)  A’s/B’s context:

\[
\begin{array}{c}
\text{the doctor} \\
\rightarrow
\\
\text{U_{Doctor}(U)} \\
?\exists x F o(x), \Diamond \\
?Ty(e \rightarrow t)
\end{array}
\]

At the next stage of processing, let’s assume that B fails to find a secure substitution for the metavariable \( U \) on the subject, thus being forced to request clarification if the requirement is to be satisfied. Notice that at this point A and B’s contexts will diverge since A presumably knows who he’s referring to, i.e. has a substituend for the metavariable introduced by the definite description. Now B’s goal tree for his request for clarification is:

(25)  B’s GOAL-TREE \( T_g \):

\[
\begin{array}{c}
\text{U_{Doctor}(U)} \\
(t, x, Chorlton'(x))_{\text{Doctor}',(t, x, Chorlton'(x))} \\
?Ty(e \rightarrow t)
\end{array}
\]

The \textsc{link} transition, which accommodates an additional property (that the individual being talked about is named *Chorlton*), takes the partial tree in (24) as its context. In this context, with the pointer at the subject node, the building of a \textsc{link} relation is licensed and this is duly constructed. Now by uttering the word *Chorlton*? a new tree can be constructed for B which indeed subsumes the goal tree of (25):
Now regular anaphoric substitution allows the metavariable $U$ to be instantiated by the term $(i, x, \text{Chorlton}'(x))$, indeed essentially, as otherwise the two nodes will not be developed as involving any shared term. The result of this process will be exactly the tree in (25) and speaker and hearer context trees will be identical at this point. As illustrated here, the most recent (partial) parse tree constitutes the most immediately available local “antecedent” for fragment resolution; hence no separate computation or definition of salience or speaker intention by the hearer is necessary for such incremental fragment construal. As in P&R, the mechanism is exactly that of apposition, the building of a LINKed structure, in this case, the result of that transition in its turn being used to provide a value for the metavariable place-holder associated with the definite article.

As we saw, the hearer, B, may respond to what he has constructed during interpretation, anticipating A’s verbal completion as in (27) and (28):

(27) Conversation from A and B, to C:
A: We’re going to ...
B: Bristol, where Jo lives.

(28) A: Are you left or
B: Right-handed.

This is facilitated by the general predictivity/goal-directedness of the DS architecture since the parser is always predicting top-down goals (requirements) to be achieved in the next steps (see stage 2 of (12) or e.g. (24)). Such goals are indeed what drives the search of the lexicon (lexical access) in generation, so a hearer who shifts to successful lexicon search before processing the anticipated lexical input provided by the speaker can become the generator and take over. In all the cases of split utterances, the original hearer is, indeed, using such anticipation to take over and offer a completion that, even though grammatically licensed, i.e. fitting the predicted structure of the context tree, might not necessarily be identical to the
one the original speaker would have accessed had they been allowed to continue their utterance as in (4)-(6). From this point of view, since both speakers and hearers are licensed to operate with partial structures, speakers can start an utterance without a fully-formed intention/plan as to how it will develop (as the psycholinguistic models in any case suggest) relying on feedback from the hearer to shape their utterance:

(29)  
A: Oh. They don’t mean us to be friends, you see. So if we want to be . . .  
B: which we do  
A: then we must keep it a secret. [natural data]

Hence the assumption of underspecified partial speaker contents before the beginning of articulation allows genuine collaboration in the construction of utterances (Goodwin 1979), without necessarily having to resort to revision and backtracking.

4 Combining Dynamic Syntax with TTR

DS doesn’t incorporate a notion of dialogue act type (in contrast to e.g. Ginzburg et al. (2003)) as it is assumed that the linguistically provided information is highly underspecified, namely just an indication of sentence mood as declarative, interrogative, imperative. However, as the DS formalism is designed to interact with context incrementally at any point, the possibility of deriving speech act information from context exists, although the interface to enable this must be specified. For this reason, we now turn to TTR, a transparent representation format allowing the specification and interaction of multiple types of information.

4.1 Type Theory with Records (TTR)

TTR has already been used in dialogue modelling Ginzburg and Cooper (2001), Ginzburg (forthcoming). Tokens (records) and types (record types) are treated uniformly as structured representations – sequences of label : type pair fields – with the result that their interaction can be modelled in a single system, notably as required when dealing with metacommunicative uses of language (in Track 2) such as ‘repair’-constructions or grounding.

Here, the attraction of TTR is that it allows the stratification of multiple types of information, using distinct field labels. The device of dependent types allows linking of information between fields, as types can depend on types occurring earlier.

11Such specifications are currently encoded as features translatable into use-neutral procedural instructions, unless there are “grammaticised” associations between moods and speech acts, an empirical issue to be decided on a language-by-language basis.
in the record (higher up in the graphical representation). This allows us to separate contextual information (e.g., information about conversational events, including speaker, addressee, time, location etc.) from the semantic content directly derived from the linguistic string, but allow interaction between the two; this is what we need for phenomena like resolution of ellipsis or assigning values to indexicals and anaphoric elements.

4.1.1 Using TTR in DS

TTR has not, however, been defined in an incremental manner. Here, then, we use TTR representations within the DS vocabulary of trees and actions, replacing the unstructured content of the \( F_0() \) labels with TTR record types, and interpreting \( Ty() \) simple type labels (and requirements) as referring to final TTR field type. Compare the modified lexical entry and eventual tree representation below with the ones displayed in (7)-(8):

\[
\begin{align*}
\text{john:} & \quad \text{IF} \quad ?Ty(e) \\
\text{THEN} & \quad \text{put}(Ty(e)) \\
\text{ELSE} & \quad \text{put}([x : john]) \\
\text{ELSE} & \quad \text{abort}
\end{align*}
\]

(30)

```
IF $Ty(e)$
THEN put($Ty(e)$)
ELSE put([x : john])
ELSE abort
```

(31) “John arrived”

```
\diamond Ty(t), \left[x : john \atop p : \lambda \left[x : e \atop p : \lambda \left[x : e \atop p : arrive(x) \right]\right]\right]
```

Function application and type deduction will now apply under a suitably modified rule of the DS Elimination process; see (33) below.

5 Utterance Events

An account of SUs must explain how indexical pronouns can assume distinct values around a change of speaker (and addressee). We therefore require some record of the utterance event/situation which includes information about speaker/addressee identity. Note that the availability of utterance events to the semantics is independently motivated by e.g., event reference via anaphora (“what do you mean by that?”) (see also Poesio and Rieser 2010).

\[12\] Work is underway to introduce incrementality in the TTR model via the subtyping relation (White (in prep); Meyer-Viol (in prep)). Here we pursue a more conservative strategy.
We assume that utterance events should at minimum record participant information and who is uttering which particular word(-string). We therefore introduce a partition within the TTR representation of content, with utterance event information held in a context (or ctxt) field, and linguistically derived semantic content in a content (or cont) field. The ctxt field is itself structured, containing the required information about utterance event, speaker and addressee; we assume this is available directly from the real-time context of utterance:

\[
\begin{align*}
  a &: \text{participantA} \\
  b &: \text{participantB} \\
  u &: \text{utt-event} \\
  s_a &: \text{spkr}(u,a) \\
  s_a &: \text{addr}(u,b)
\end{align*}
\]

(32)

In a fuller treatment, this utterance context information should also include further information such as time of utterance, world etc, but we omit these here for simplicity.

The DS Elimination process must now perform beta-reduction (as before) for the cont field, and TTR extension (i.e. concatenation Cooper (1998), shown here as \(\oplus\)) for the ctxt field, as shown in (33), (34):

\[
\text{Elimination:} \\
\text{IF} \quad ?T_y(T_1), \\
\quad \downarrow_0 (T_y(T_2), \begin{bmatrix} \text{ctxt} : c_1 \\ \text{cont} : \alpha \end{bmatrix}) \\
\quad \downarrow_1 (T_y(T_2 \rightarrow T_1), \begin{bmatrix} \text{ctxt} : c_2 \\ \text{cont} : \beta \end{bmatrix}) \\
\text{THEN} \quad \text{put}(T_y(T_1), \begin{bmatrix} \text{ctxt} : c_1 \oplus c_2 \\ \text{cont} : \beta(\alpha) \end{bmatrix}) \\
\text{ELSE} \quad \text{abort}
\]

\[
\Diamond, T_y(t), \begin{bmatrix} \text{ctxt} : c_1 \oplus c_2 \\ \text{cont} : \begin{bmatrix} x : e \\ p : f(x) \end{bmatrix} \end{bmatrix}
\]

(34)

\[
\begin{align*}
  T_y(e), \begin{bmatrix} \text{ctxt} : c_2 \\ \text{cont} : \begin{bmatrix} x : e \end{bmatrix} \end{bmatrix} &\quad \quad T_y(e \rightarrow t), \begin{bmatrix} \text{ctxt} : c_1 \\ \text{cont} : \lambda x : e . \begin{bmatrix} x : e \\ p : f(x) \end{bmatrix} \end{bmatrix}
\end{align*}
\]

\[\text{13This is a simplification, of course: determination of addressee is not trivial – see Goffman (1981) amongst others.}\]
Parsing a two-word utterance, e.g., *John arrived*, spoken by one speaker, A, will therefore now result in a representation as in (35):

\[(35) \quad \diamond, Ty(t), \begin{cases} \text{ctxt} : & \begin{array}{l} a : \text{participant}A \\ u_0 : \text{utt} - \text{event} \\ s_{u0} : \text{spkr}(u_0, a) \\ u_1 : \text{utt} - \text{event} \\ s_{u1} : \text{spkr}(u_1, a) \end{array} \\ \text{cont} : & \begin{array}{l} x : \text{john} \\ p : \text{arrive}(x) \end{array} \end{cases} \]

\[(36) I: \begin{cases} \text{IF} & ?Ty(e), \begin{cases} \text{ctxt} : & \begin{array}{l} s_a : \text{spkr}(u, x) \end{array} \end{cases} \\ \text{THEN} & \begin{array}{l} \text{put}(Ty(e)) \\ \text{put}(Fo(x)) \end{array} \end{cases} \\ \text{ELSE} & \text{abort} \end{cases}\]

\[(37) You: \begin{cases} \text{IF} & ?Ty(e), \begin{cases} \text{ctxt} : & \begin{array}{l} s_a : \text{addr}(u, x) \end{array} \end{cases} \\ \text{THEN} & \begin{array}{l} \text{put}(Ty(e)) \\ \text{put}(Fo(x)) \end{array} \end{cases} \\ \text{ELSE} & \text{abort} \end{cases}\]

\[14\text{A more complex set of actions may be required to account for the fact that you may be singular or plural in reference, may include the hearer or not and may be generic.}\]
As grammatical constraints in DS are phrased in terms of semantic features (rather than syntactic features), the grammaticality of examples like (3) now becomes almost trivial. While a syntactic account would have trouble explaining how myself can be co-referential with its antecedent you, there is no such problem here: you uttered by A and myself uttered by B annotate the trees with co-referential semantic variables. The lexical entries for reflexives such as myself must check for a suitably co-referential subject elsewhere in the tree (via the co-argument constraint $\uparrow_0\uparrow_1\downarrow_0\mathrm{Fo}(x)$), and here, this will be available:

(38) \textit{myself:}
\begin{align*}
\text{IF} & \quad ?Ty(e), [\quad ctxt : [s_a : spkr(u, x)]], \\
& \quad \uparrow_0\uparrow_1\downarrow_0\mathrm{Fo}(x) \\
\text{THEN} & \quad \text{put}(Ty(e)) \\
& \quad \text{put}(\mathrm{Fo}(x)) \\
\text{ELSE} & \quad \text{abort}
\end{align*}

6 Speech acts

Purver et al. (2009a) show that SUs are often not straightforward in speech act terms: sometimes they continue/complete the original speech act; sometimes they perform a new one, clarifying/confirming a suggested completion; sometimes they are ambiguous and/or multifunctional. In order to express these important differences, we need the ability to represent and reason about speech act information (see e.g. Ginzburg et al. (2003), Asher and Lascarides (2003)).

Importantly, we would like any inferences about speech acts to be \textit{optional}. A parser should enable these inferences when the appropriate function of the turn is at issue (e.g. in cases of ‘repair’), but they should not have to be derived for intelligibility or the determination of grammaticality. They should also be derivable retrospectively: as a result of an interlocutor’s feedback, one can assign a particular force (even to one’s own contribution) that had not occurred to them beforehand.

Any computational rules that introduce such inferences must therefore be available in the grammar but optional (except where the association of a specific construction with a particular interpretation has been grammaticised); and the resulting representations should be kept distinct from those derived directly from the parsing of linguistic input. DS already provides a mechanism which suits these requirements: the use of LINKed trees (trees which share some semantic variable), as in the analysis of non-sentential fragments Gargett et al. (2009) and relative clauses Kempson et al. (2001a). This device of LINKed trees expresses the cognitive reality of distinguished local domains as evinced by standard syntactic tests, e.g. island-constraints and binding restrictions (see e.g. Gregoromichelaki (2006)).
As TTR currently does not provide the means for such syntax-semantics interface restrictions we retain the notion of LINKed trees here.

As speech act information can be highly underspecified and context-dependent, we do not wish to assume here either a fixed range of speech acts or a fixed set of inferences from linguistic form to speech act type. We therefore take the rules introducing such information to be of the form sketched in (39). When applied, this rule will introduce a new LINKed tree and provide a \( Fo \) value \( A(V, U, F(p)) \) where \( A \) is a metavariable ranging over speech act specifications, \( V \) the agent responsible for the speech act, \( U \) an utterance event (or sequence of events), and \( F \) some function over the semantic content of the utterance (\( p \) and \( x \) are rule-level variables binding terms on the nodes where the rules apply):

\[
\text{IF } Ty(x), Fo(p) \text{ THEN make}(L), go(L) \]  
\[
\quad \text{put}(A(U, V, F(p))) \]  
\[
\text{ELSE abort} \]

In order to distinguish content that is derived directly on the basis of linguistically provided information and content derived on the basis of such inferences we introduce a partition in the TTR representation: we take the \textit{cont} field to indicate the (linguistically-derived) truth conditional content and introduce an \textit{inf} field for the speech act content derived by means of such rules (this roughly corresponds to the \textit{explicature/high level explicature} distinction in Relevance Theory). So, for illustration, a suitable (optional) rule for assertions might perhaps apply to \( Ty(t) \) trees with proposition \( p \) and speaker \( a \), allowing one to infer the extra content \( assert(a, p) \):

\[ ^{15} \text{The nature of } F \text{ will depend on speech act type; for an assertion, it may simply be the identity operator; for irony, negation (see e.g. Asher and Lascarides (2003) for suggestions on how speech act type may relate to semantics).} \]
Given this, we can outline an account of SUs in which the same linguistic input can be construed as performing different possible speech acts (perhaps simultaneously). Consider the simple (and constructed) example in (41):

(41) A: John . . .
    B: arrived?

There are (at least) two possible readings of the resulting collaboratively produced contribution: one in which B is (co-)querying whether John arrived; and one in which B is clarifying A’s original speech act, i.e., B is asking whether A was asking whether John arrived.

The tree resulting from parsing (or producing) this SU will be similar to the one in (35) above, except that, due to the speaker change, the second utterance event $u_1$ is shown as spoken by B (see the unboxed part of Fig 1).

Applications of computational rules as in (39) above allow us to infer the speech act information corresponding to the two possible readings, deriving LINKed sub-trees which indicate speech acts performed by whichever participant is taken as the agent. One possible inference would derive the simple collaborative “co-querying” reading (based on the interrogative intonation and the identity of the final speaker B who is completing A’s question) adding the speech act proposition that B and A are asking (potentially an audience C, not shown here) whether John arrived – see the upper box in Fig 1. An alternative inference would derive the “clarificational” reading shown in the lower box. Of course, other inferences may
Figure 1: SU-derived tree
also be possible.\textsuperscript{16}

Note that (35) and Fig 1 display representations of the final state that a parser might be in after B’s contribution; from an incremental processing point of view, we are also interested in the state at the transition point (the change in speaker). Without considering any speech act inference, the tree at this transition point will be as follows:

\begin{equation}
Tn(0), \, ?Ty(t)
\end{equation}

\begin{equation}
\begin{aligned}
& Ty(e), \\
& \begin{cases}
  ctx : \left[ u_0 : \text{utt} - \text{event} \\
  s_{x_0} : \text{spkr}(u_0, a) \\
  cont : \left[ x : \text{john} \right]
\end{cases}
\end{aligned}
\end{equation}

\begin{equation}
?Ty(e \rightarrow t), \, \Diamond
\end{equation}

This tree is partial (i.e. incomplete, having as yet unsatisfied requirements), but in itself is enough for B to begin generating – provided that they have some suitable message in mind (encoded as a \textit{goal tree} in DS) which is subsumed by this partial tree. There is no requirement for B (or indeed A) to complete this tree, or perform any inference about speech acts, in order to begin generation (or, in A’s case, parsing). In cases where B’s continuation matches what the original speaker A could have intended to convey, the appearance would be one of “guessing”, even though B has not performed any kind of inference regarding A’s speech act. In fact, as (4), repeated below, shows, completions of another speaker’s utterance by no means need to be what the original speaker actually had in mind:

\begin{equation}
\text{(43) Daughter: Oh here dad, a good way to get those corners out} \\
\text{Dad: is to stick yer finger inside.} \\
\text{Daughter: well, that’s one way.} \\
\text{[from Lerner (1991)]}
\end{equation}

Such continuations can be completely the opposite of what the original speaker might have intended as in what we will call “hostile continuations” or “devious suggestions” – which are nevertheless collaboratively constructed from a syntactic point of view:

\begin{equation}
\text{(44) (A and B arguing:)}
\end{equation}

\textsuperscript{16}If such inferences become grammaticised, i.e. a particular construction is associated with a particular act (e.g. \textit{clarification}), only one rule may be available. This is an empirical issue which we set aside here, but see Ginzburg (forthcoming).
A: In fact what this shows is
B: that you are an idiot

(45)  (A mother, B son)
      A: This afternoon first you’ll do your homework, then wash the dishes and
      then
      B: you’ll give me £10?

Given a suitable model of the domain at hand, B, sometimes, will presumably be
able to determine the content of A’s intended speech act and represent it as such,
i.e., as a speech act emanating from A, in their goal tree (see e.g. Poesio and
Rieser (2010)). We take this not to be an essential process for the production of
SUs, although it could be necessary in cases where B’s next move is specifically
intended as a confirmation request for such a representation.

6.1.1 The Split Turn-Taking Puzzle

Ginzburg (1997) describes a Turn-Taking Puzzle (TTP), which, he argues, shows
that options for ellipsis resolution are distinct for speaker and hearer. This is illus-
trated by means of why-fragments:

(46)  A: Which members of our team own a parakeet?
      B: Why? (= ‘Why are you asking which members of our team own a
      parakeet?’)

(47)  A: Which members of our team own a parakeet? Why?
      (a) = ‘Why own a parakeet?’
      (b) # ‘Why am I asking this?’

(48)  A: Which members of our team own a parakeet? Why am I asking this
      question?

According to Ginzburg, the reading in which why queries the intended speech act
(the why meta reading) is available when asked by B as in (46) but unavailable when
asked by the original speaker A as shown in (47b). However, this cannot simply be
due to coherence or plausibility, as this reading is available in (48) when expressed
by non-elliptical means. Its unavailability must therefore be related to the way
context is structured differentially for speaker and hearer.
Our explanation of this puzzle takes the $why_{\text{meta}}$ interpretation as querying the intention/plan\(^{17}\) behind the original speaker’s speech act.\(^{18}\) Since ellipsis resolution requires the potential for immediate accessibility of a salient representation, the unavailability of the reading in (47b) shows that the speaker’s own intention behind their speech act is, in general, not salient enough for them to question it through $why$-ellipsis\(^{19}\) (in Ginzburg’s formulation such a fact does not belong in the TOPICAL FACTS field; however, this fact obtaining is not impossible, as (48) shows). Under this explanation, the TTP then reveals which agent takes responsibility for performing the relevant speech act, and hence can be queried about their intentions behind this act. In terms of Goffman (1981)’s distinctions among “speaker”-roles, the relevant agent is the ‘Principal’. This distinction can be clearly evident in cases of SUs in multi-party dialogue. Now the utterer of a completion (the final “speaker” in the general sense discussed so far, and as indexed by pronouns like $my$) can felicitously ask elliptical $why_{\text{meta}}$ questions of the original speaker (we will call this phenomenon the STTP, or Split Turn-Taking Puzzle):

(49) A to C: Have you finished sharpening ...
B to C/A: my loppers? B to A: Why?
   (a) = ‘Why are you asking C whether she has finished sharpening my loppers?’
A to B: Because I want her to sharpen my secateurs too.

We can explain B’s $why$-fragment interpretation in (49a) if we assume that although B’s fragment $my$ loppers? completes A’s question for C (in addition, it could be clarificatory for A), B does not necessarily assume responsibility for the performance of the speech act. That is, A must be taken as the agent of the querying speech act even though there is a sequence of utterance events which A and B have performed severally with B the final utterer.\(^{20}\) The availability of the $why_{\text{meta}}$ reading then follows, even though apparently in contrast to (47b).

In some cases, then, even though the turn is collaboratively constructed, the original utterer maintains the authority or responsibility for the turn even though it

---

\(^{17}\)Note that this approach does not necessitate that speech act and therefore intention information is available PRIOR to the processing of the $why$-question: instead, seeking to interpret such questions can be the trigger for optional (speech-act inducing) rules to apply. Hence, this approach is perfectly compatible with the general view on intentions as post-facto constructs (see e.g. Suchman (1987/2007)) and the fact that conversational participants negotiate the content of speech acts with such assignments able to emerge retrospectively.

\(^{18}\)As Ginzburg (forthcoming) notes, recognition of this intention is not necessary for grounding.

\(^{19}\)However, it is not impossible:


\(^{20}\)In fact, the specification of the $why$-fragment as $why_{\text{meta}}$ can be taken to trigger the inference that A is solely responsible for the query as B now dissociates himself from it.
was completed by somebody else. In other cases, see e.g. the hostile completions (44) and devious suggestions (6), this is not the case: the eventual content derived has to be taken as solely attributable to the second speaker. Notice however that in all cases (except those of direct quotation), the content of indexicals like *my* and *you* tracks directly the actual utterer/listener, irrespective of who is taking responsibility for the content (or speech act performance). Even in helping out somebody to finish their sentence such indexicals will track the actual utterer/listener:

(50) Child (playing with toy garden tools): Give me my . . . 
   Mum: your secateurs. Here they are, in fact these are loppers.

(51) A: Next cut it with your ... 
   B: my loppers. No, this we cut with the secateurs.

This provides evidence for the dissociation of speech act performance and performance of the utterance event: these are two distinct actions whose agent might coincide but not necessarily so (these two roles roughly correspond to Goffman (1981)’s ‘Author/Principal’ and ‘Animator’). Most accounts conflate the two: Las- carides and Asher (2009) argue that each time a speaker makes a conversational move they undertake a *public commitment*. However, SU examples such as (2)-(3) and (50)-(51) show that the person undertaking the public commitment (the ‘Principal’) does not necessarily coincide with actual utterer (the ‘Animator’). We therefore conclude that the notion of ‘commitment’ should be correlated with something else, namely, who is performing (the agent of) the associated speech act (which could be the two speakers jointly but not necessarily and not only for SUs). Speech act inference rules as outlined in (39) must therefore maintain the flexibility to assign the inferred speech act to any of the speakers involved, and not only the final one.

7 Conclusions

The STTP and the multifunctionality of SU fragments motivates our claim that information manipulated during a parse has to be structured in a uniform representation that includes (at least) three levels that can make reference to each other: (a) semantic content which is directly derived on the basis of the linguistic input (but with additional inference), (b) context specifications arising from the utterance situation (utterance events) and (c) optional speech act information. All three types of information can interact and influence the derivation of each other as seen above. Formulation of this information in a DS-TTR combined formalism allows the precise specification of the interactions required for the appropriate processing of SUs and the formulation of constraints in the derivation of their import in dialogue.
Hence the type of phenomenon examined here, SUs and their interaction with
the structuring of context and purely grammatical/linguistic information (e.g. in-
dexicals), indicates that the standard dichotomies between grammar/pragmatics,
official/unofficial business of conversation or competence/performance models needs
reconsideration. Given the incrementality of the relevant mechanisms assumed
here, the interaction between these components can be shown to be essential in
their stepwise functioning hence requiring uniform modelling under a single rep-
resentational system. More broadly, the view that emerges entails a reconsider-
ation of ‘signalling’ (or ‘ostensive communication’) as reliant on highly struc-
tured contextual representations/mechanisms and a non-individualistic view on
‘meaning’ (Millikan 1993, 2005) as jointly constructed in interaction. Coordina-
tion/alignment/intersubjectivity among dialogue participants, instead of requiring
metarepresentational abilities (e.g. mind-reading) can then be seen as the product
of low-level mechanisms (Pickering and Garrod 2004; Mills and Gregoromichelaki
2010, Böckler et al 2010, Gregoromichelaki et al 2011) like the grammar (con-
ceived appropriately as part of the mechanisms underpinning a general representa-
tional capacity).

Acknowledgments

We gratefully acknowledge valuable support and advice from Arash Eshghi, Chris
Howes, Pat Healey, Greg Mills, Graham White. This work was supported by Dyn-
Dial (ESRC-RES-062-23-0962).

References

Nicholas Asher and Alex Lascarides. *Logics of Conversation*. Cambridge Univer-


Ronnie Cann. Towards an account of the english auxiliary system. In R. Kempson,
Gregoromichelaki E., and C. Howes, editors, *The Dynamics of Lexical Inter-

Ronnie Cann, Ruth Kempson, and Lutz Marten. *The Dynamics of Language*. El-


