Amstelogue'99 Proceedings

Workshop on the Semantics and Pragmatics of Dialogue
Amsterdam University 7-9 May 1999

Part I

editors:

Jan van Kuppevelt
Noor van Leusen
Robert van Rooy
Henk Zeevat

sponsors:

DISC(Esprit)
Dutch Research Foundation (NWO)
Faculty of Humanities (University of Amsterdam)
Institute for Logic, Language and Information
Dutch Research School on Logic (OZSL)
TRINDI(Language Engineering)

Preface

This volume collects most of the papers presented at the Amstelogue workshop in May 1999. They have all been refereed but we have decided against asking for new versions to prevent further delay. A number of authors will be asked to resubmit for the special Amstelogue issue of the Journal of Semantics.

Our thanks go to all speakers, referees and contributors for making the Amstelogue workshop and enjoyable and fruitful event. We further thank the Faculty of Humanities, the University of Amsterdam, the Institute for Logic, Language and Computation, the Research School in Logic, the Dutch Research Foundation, the DISC project and the TRINDI project for their financial contributions.

Jan van Kuppevelt
Noor van Leusen
Robert van Rooy
Henk Zeevat

List of Contents

Part I

Part II

13. Hannes Rieser
The Structure of Task-oriented Dialogue and the Introduction of New Objects
email:rieser@lili.uni-bielefeld.de

14. Robert van Rooy
Permission to Change
email:vanrooy@hum.uva.nl

15. David Traum
20 Questions on Dialogue Act Taxonomies
email:traum@cd.umd.edu

16. Claus Zinn
Understanding Mathematical Discourse
email:zinn@informatik.uni-erlangen.de

Truth Conditional Discourse Semantics for Parentheticals*

Nicholas Asher
Department of Philosophy
The University of Texas at Austin

Abstract

In this paper I propose a truth conditional analysis of parentheticals and discourse adverbials that extends also to an account of so called pragmatic conditionals. I argue that a proper account of these elements involves an appeal to a complex notion of discourse structure such as that developed in SDRT.

1 Introduction

There is a tradition in pragmatics going back at least to Grice according to which certain parts of speech don't contribute to the truth conditional content of the assertions of which they are part. Rather they implicate or indicate either a particular speech act or an attitude of the speaker. Examples of such items are:

- mood indicators—questions, commands.
- interjections—*Oh, Gee, Too bad, Damn,* etc.
- so called *discourse adverbials—allegedly, unfortunately*, etc. This category also includes adverbial clauses—e.g., *as Mary assures us.*
- so called *pragmatic conditionals—if you know what I mean, if you see what I'm getting at.*
- discourse particles—*tte* in Japanese or *re* in Sissala for hearsay.
- discourse connectors—*but, too, hence, so, therefore*, etc.
- parenthetical constructions, in which full clauses missing a verbal complement occur.

*I would like to thank Lilliane Haegeman, Alex Lascarides and Frank Veltman for helpful comments.

Wilson and others have argued that all of these phenomena exhibit a similar behavior relative to a test for non-truth-conditional meaning—the "embedding test." Sperber and Wilson (1995) also claim that they can give a unified analysis of these phenomena. However, I'll argue here that the test does not really separate out parts of speech with a non-truth conditional meaning (whatever that might be is not my concern here). Certainly, there is reason to doubt that all of these constructions fail to be amenable to truth conditional or, more generally, model theoretic analysis. Some mood indicators have received detailed and rigorous model theoretic analysis in e.g. Hintikka (1974) and Groenendijk and Stokhof (1984). Others have argued that discourse connectors have an important though sometimes subtle effect on the truth conditional interpretation of discourse (Asher 1993, Lascarides and Asher 1993) and that hence traditional pragmatics gives a misleading picture of discourse interpretation by separating out the contribution of discourse connectors from an account of truth conditional content. Similar remarks apply to the Japanese discourse particle *tte* (Hasegawa 1996).

In this paper I will examine parentheticals and discourse adverbials. Some examples of parentheticals are given in (1) below (the parentheticals are underlined). I'll argue that these parts of speech also have a straightforward truth conditional semantics in a theory of discourse interpretation that takes account of discourse structure. The theory of discourse interpretation that I will use is SDRT (Asher 1993, Lascarides and Asher 1993), an extension of DRT that incorporates an account of discourse structure and rhetorical function.

(1) a. The party is over, <u>I hear</u>.
b. Please leave, <u>I beg you</u>.
c. The party, <u>Mary assures us</u>, is over.

2 A test for non truth conditional meaning?

According to Wilson (1975), there is a test for non-truth-conditional meaning: embed the questionable item into the antecedent of a conditional and see if the purported truth conditional contributor's meaning falls within the scope of 'if'. If it does, it's truth conditional; and if not, not. Here are some examples of the test at work:

(2) a. If the party, unfortunately, is over, then we should find somewhere else to get a drink.
?⟶
If it is unfortunate that the party is over, then we should find somewhere else to get a drink.
b. If the sun is shining but it's midnight, then we must be in Norway.
?⟶
If the sun is shining and it's midnight and that's not expected, then we must be in Norway.
c. If, I'm warning you, you cross that line, I'll hit you.
?⟶
If I'm warning you that you cross that line, I'll hit you.

According to the test, these examples appear to indicate that neither parentheticals, discourse adverbials, nor discourse particles have a truth conditional import, because their supposed content can't embed inside a conditional. But this conclusion is too hasty. According to this test, nonrestricted relative clauses and appositive NPs would fail to have a truth conditional import, when they obviously do:

(3) If the party, which Jane attended, is over, then we should find some where else to get a drink.
$? \longrightarrow$
If the party is over and Jane attended that party, then we should find somewhere else to get a drink.

(4) If the party, that one that Jane is hosting, is over, then we should find somewhere else to get a drink.
$? \longrightarrow$
If the party is over and Jane is hosting that party, then we should find somewhere else to get a drink.

Crucially, what's wrong with the test for non truth conditional meaning is that it overlooks the obvious possibility that the content of the apparently non-truth-conditional item may simply fall outside the scope of the conditional but nevertheless contribute to the truth conditions of the discourse.

Before dismissing this test, it is nevertheless important to note that it *does* render very dubious an account of parentheticals as syntactically displaced constituents. Such a simple account of parentheticals would see the examples in (1) as equivalent to sentences in which the main clause is a complement to the expression in the parenthetical. Thus, (1c) would be equivalent to

(5) I hear that the party is over.

While this is plausible for such simple sentences, once the parenthetical occurs within a clause of a complex sentence like those used in the embedding test, this simple syntactic account makes the wrong predictions. Further convincing, syntactic evidence that parentheticals, discourse adverbials and interjections remain unattached "orphans" at syntactic structure has been given by Haegeman (1991) (see also Haegeman 1984). Thus, if we are to provide a unified interpretation for sentences containing parentheticals or discourse adverbials, we will have to move to a semantic or even pragmatic-semantic account of logical form, which is what I turn to now.

3 A Positive Account

Parentheticals and discourse adverbials share several features with presuppositions. First, both typically project out of the context in which they are introduced. Projection means that presuppositions also fail the "test" for truth conditional meaning propounded by Wilson:

(6) If the King of Buganda is bald, then he wears a wig in public. $? \longrightarrow$ If there is a King of Buganda and the King of Buganda is bald, then he wears a wig in public.

Second, both parentheticals and discourse adverbials on the one hand and presuppositions on the other typically convey propositions, once certain anaphorically underspecified elements are resolved. Asher and Lascarides (1998) present an SDRT account of presuppositions according to which presuppositions must be attached to some antecedent, available part of the discourse context *via* a restricted range of discourse relations that represent their discourse function. These two relations are Background, in which the presupposition gives some stage setting information about element in the main narrative line of the text, and Defeasible Consequence, in which the presupposition is a defeasible consequence of the constituent to which it is related. One of these two relations always attaches a presupposition to the discourse context, unless the presupposition trigger itself specifies a discourse relation (e.g., the presupposition trigger *too* introduces the discourse relation Parallel, as argued in Asher 1993). Following van der Sandt's account, the SDRT account of presupposition supposes that presuppositions that can't be derived from or "bound to" the context (and so attached with Defeasible Consequence) prefer an attachment to as superordinate a position as possible in the discourse context.

Parentheticals, like presuppositions, are anaphoric and express propositions. In SDRT they must attach to some part of the discourse context via a discourse relation. Some parentheticals—viz. the epithets, expressions like *he commented*, and many discourse adverbials—determine a particular discourse relation like Commentary. Others encode their rhetorical functions in their main verbs: *Mary explained, Mary elaborated, Mary replied.* Parentheticals prefer a different attachment to presuppositions: they typically attach to a discourse constituent formed from the asserted clause or sentence in which they are embedded, whereas presuppositions can attach at any available position and even prefer high attachment with Background (the counterpart to van der Sandt's rule of wide scope accommodation). Consider for instance,

(7) John, Mary assures us, can be trusted.

I'll assume that syntax can isolate out the parenthetical and lexical analysis reveals that verbs used in our parenthetical examples like *assure, hope, fear* as well as sentence adverbials like *unfortunately* have an argument that needs to be specified. Thus, we would have for (7) something like the following logical form or SDRS:

$$(8)\quad \begin{array}{|l|}\hline \pi', R, v \\ \hline \pi : \begin{array}{|l|}\hline m, X, p \\ \hline \text{assures}(m, X, p) \\ p = \textit{?} \\ \hline \end{array} \\ R(v, \pi) \\ R = \textit{?} \\ v = \textit{?} \\ \hline \end{array}$$

In (8) π labels the constituent derived from the parenthetical material, p stands for the object of *assure* that has yet to be specified, R stands for a yet to be specified discourse relation and v is the label of a yet to be specified attachment point for the parenthetical constituent. Compositional semantics gets us a logical form for the assertion minus the parenthetical—which I'll label with π'. The rule for processing parentheticals resolves the underspecified condition $v = \textit{?}$ to $v = \pi'$, while the resolution of the underspecified conditions $p = \textit{?}$ and $R = \textit{?}$ is left to independent processes of anaphora resolution and SDRT's computation of discourse relations respectively. In this case, there is only one proposition in the context to identify with p—the proposition that John can be trusted. And the relation R can be specified given the content of the parenthetical to Evidence. On this analysis, we have a pretty straightforward truthconditional analysis of parentheticals.

My account is slightly more complicated than one might think necessary. Why must we, one might argue, attach the parenthetical with a discourse relation to some constituent? Couldn't we just assume that p is always identified with the entire surrounding assertion, thus reviving the syntactic analysis at the level of logical form? On this simpler proposal, we would predict (7) to be equivalent to:

(9) Mary assures us that John can be trusted.

This account could even be used to get wide scope readings of the parenthetical material for the embedding test sentences. Thus, something like (2c) would be equivalent to:

(10) I'm warning you that if you cross the line, I'll hit you.

On the other hand, these predicted equivalences run into trouble with slightly more complex sentences. Consider,

(11) a. Mary assures us that John can be trusted, but I don't trust him.
b. John, Mary assures us, can be trusted, but I don't trust him.

Informants find (11b) odd. And the reason, I think, is that the use of certain parentheticals that use evidential verbs like *assure, swear, testify* and *affirm* commit the speaker to the truth of the attachment point for the parenthetical.[1] The simple non-discourse based story can't make such a difference between (11a) and (11b).

Another difference between the simpler account and my discourse based account is that the latter predicts better when some parentheticals *don't work.* Consider:

(12) a. Please leave, I'm begging you.
b. You must leave, I'm begging you.
c. ?You must be tired, I'm begging you.
d. ?The party must be over, I'm begging you.

A purely semantic (or syntactic) account must either accept (12bcd) or reject them all. All of these parentheticals have the same sort of semantic object, a proposition, which I argued in Asher (1993) isn't quite the right sort of object for buletic verbs like *beg* or *want.* But because on this account parentheticals have an anaphorically specified argument, we can speculate that some coercion there is allowed (especially when the indirect object of the parenthetical coincides with the subject of the main clause). Further, the rhetorical connection helps the coercion as well. As a result, (12b) is perfectly fine; there's a clear rhetorical relation between begging someone to leave and the person's having to leave. Having to leave is a natural end for a beggind to leave; there 's a natural Narrative link between the two.. For (12c), however, the rhetorical relation isn't at all clear, nor again for (12d. Or at least you'd need a particular context where someone's begging you to be tired makes you so. The non-discourse based story cannot account for the differences between (12b) and (12c) in any way, as far as I can see.

The discourse based account sketched leaves open exactly what an appropriate attachment point for the parenthetical is. In the simple example (7) above, the entire assertion served as attachment point. What about more complex sentences? On the one hand, we have wide scope readings of parentheticals.

(13) a. Only if, I fear, we work like dogs, will we be able to save this company.

b. Even if, I hear, the reception's over, we'll still be able to get something to eat.

c. If, I'm warning you, you cross that line, I'll hit you.

These examples all seem to invite at least the reading on which the parenthetical's anaphoric argument p is identified with the entire assertion surrounding the parenthetical.

On the other hand, we have examples in which the parenthetical has scope over the consequent, at least according to one informant.

(14) If the party, unfortunately, is over, then we have to go somewhere else to get a drink.

There most salient interpretation of (14) is that if the party is over, then we should find somewhere else to get a drink and it is unfortunate that we should find somewhere else to get a drink. This attachment is one that's quite different from a presupposition, since the parenthetical is attached and resolved to some *non-accessible* element in the discourse structure (following van der Sandt). But it's a matter of cataphoric resolution of an underspecification, which should be admissable insofar as other cataphoric links are admissable.

Finally, we have certain examples of attachment of parenthetical material or discourse adverbials to the antecedent of a conditional. Consider the following three examples.

(15) a. If the party, as Mary assures us, is over, then we should find somewhere else to get a drink.

b. If the party, unfortunately, is over, then we should find somewhere else to get a drink.

c. If the party, unfortunately, is over, then we should go home.

In (15a) the parenthetical takes the antecedent of the main assertion in its scope and attaches with Background to the conditional. The most salient reading of (15a) is: Mary assures us that the party is over, and if the party is over, then we should find somewhere else to get a drink. (15b) gives rise to two interpetations: (i) it's unfortunate that if the party is over, then we should find somewhere else to get a drink; (ii) if the party is over, then that's unfortunate and if the party's over we should find somewhere else to get a drink. The first reading is a straightforward application of the account given so far, but what about the second? In fact the second seems more plausible.

To account for these various readings, we need to appeal in greater detail to the nature of discourse relations and attachment points in SDRT. The wide scope reading of (13a) can be understood as a Commentary on the assertion itself.[2] On this reading, (13a) is equivalent to:

(16) Only if we work like dogs, will we be able to save this company and, as a Commentary, I fear that only if we work like dogs, will we be able to save this company,

This reading follows from our decision to treat the parenthetical as a real discourse constituent. There is perhaps also a narrower scope reading for (13a) where the Commentary extends only over the antecedent of the conditional–more on this below.

The other two examples of widescope readings don't generate a Commentary relation in the attachment reasoning process. The parenthetical *I hear* generates an Evidence relation in the widescope reading for (13b) and is analyzed in a manner as (13a). But now what about (13c)? How does the warning that if you cross that line I'll hit you relate to the assertion that if you cross that line I'll hit you? The warning, it seems, has the assertion as a Result. Warnings are factive!

The attachment possibilities for the parenthetical reflect the resolution of the anaphoric element. In all of the examples (13), (14) and (15), the attachment point and the constituent identified as the complement of the parenthetical or discourse adverbial coincide. Why should this be? In general it is because of our relations Commentary and Evidence. In SDRT attachments are decided so as to maximize discourse coherence. This principle, `Maximize Discourse Coherence`, can be stated informally:

- `Maximize Discourse Coherence:` In updating a discourse context τ with new information ϕ, resolve all those underspecifications not resolved by the glue logic for discourse relations or by monotonic constraints on lexical choice and logical form that lead to the production of an update that is τ, ϕ maximal.

A τ, ϕ maximal update is one in which a maximal number of underspecified elements have been resolved and in which each discourse relation in the structure is as strong as it can be. Underlying this principle is the idea that many discourse relations are scalar—e.g. Parallel, Contrast, Explanation, and, especially for our purposes, Commentary and Evidence. These relations are maximally strong when we give evidence for or commentary on the *whole* of the constituent that is the attachment point. And this comes about precisely when we resolve the anaphoric element p in the representation of the parenthetical to the constituent that is the attachment site.

This leaves open the question as to why the attachments and anaphoric resolutions are chosen the way they are. This too is sometimes a matter of Maximize Discourse Coherence, as we shall see below. But there may also be other elements at work, like the position of the parenthetical and the intonational contour used with it. Further, negative Commentaries like *fear* or *unfortunately* will be less coherent as a rule when they are attached to what are seen as positive outcomes (e.g., turning the company around). So such attachments will be in general dispreferred. But such observations don't account for all the data.

In analyzing various attachment possibilities, let's turn first to some examples of narrow scope or local attachments in (15)— (15b-c) and their more salient second readings. There is a conflict between the status the conditional confers on the proposition that the party is over and the way the adverbial together with its resolved argument gets attached to the discourse context. On my account, the adverbial gives rise to a labeled DRS K_π containing the conditions *Unfortunate*(p) and $p = ?$; and on the reading we're interested in here, p is identified with the proposition expressed by the clause surrounding the parenthetical, and the natural, albeit defeasible, resolution of the underspecified discourse relation between K_π and the discourse context suggested by the adverbial itself is Commentary. But Commentary, like many discourse relations in SDRT, is veridical—i.e., implies the truth of the propositions associated with its terms. Formally, this translates to: Commentary$(\pi', \pi) \rightarrow (K_{\pi'} \wedge K_\pi)$. Further, *Unfortunate*$(p)$ generates the presupposition that the proposition identified with p is true. So on this way of interpreting the adverbial, we imply that the party in fact is over, but this conflicts with dependence of this constituent on the antecedent of the conditional. From SDRT's perspective the presence of a conditional operator signals a discourse relation between labels for the conditional's antecedent and consequent. I call this relation *Defeasible Consequence*, and it is nonveridical. That is, it does not entail the truth of the formulas associated with its terms. In fact, it is the *only* nonveridical discourse relation for monologue.

In any case, there is a conflict between two modes of attachment for a given discourse constituent in (15bc). The same analysis holds if we try to resolve p to the consequent of the conditional in these examples, which doesn't seem to be a very salient reading here but is for (14).

How can we attach the expression of the author's opinion then to the antecedent? SDRT resolves this conflict by overriding the defeasible preference for using Commentary with this discourse adverbial of attachingK_π to the context in favor of attaching it with the one nonveridical rhetorical relation for monologue, Defeasible Consequence, in which the suppositional, nonestablished character of the antedecent of (14) is preserved. On this way of attaching K_π, the presupposition generated by the adverbial can also be bound. In fact, we can bring forward the following generalization:

- When a preferred veridical relation cannot be used because of conflict with the nonveridical, conditional status of the chosen attachment point π, attach with Defeasible Consequence to π.

This generalization accounts for the conditional reading of the parenthetical in (15b-c).

More evidence for this analysis comes from examining why (15a) lacks a conditional reading. The reason that the conditional reading is absent in (15a), even though the parenthetical attaches to the antecedent, is that the adverbial doesn't generate a presupposition that its propositional object is true. Hence, there isn't any conflict in (15a) between the nonfactual status of the proposition that the party is over that comes from the conditional and the nonfactual status of the same proposition that comes from the adverbial. So there isn't any need for the SDRT construction procedure to override the default attachment of the constituent constructed from the adverbial to the outside context.

My analysis of the parentheticals in (15) is part of a more general strategy for resolving conflicts between veridical and non veridical discourse relations. Consider examples like the following that are related to the phenomenon of modal subordination noticed by Roberts (1986).

(17) a. If a shepard goes to the mountains (π_1), he normally brings his dog (π_2). He brings a good walking stick too (π_3).

b. If the children got a chess set from that store (π_1), it probably came with a spare pawn (π_2). Then it rolled off the table (π_3).

Both (17a) and (17b) exhibit in the first sentence some sort of conditional which generates the Defeasible Consequence, non-veridical relation between constituents. In both examples (π_3)

must be attached to the consequent of the conditionals if the anaphoric pronoun is to get an antecedent. But there's a potential conflict between the entailment of the veridical relation—Parallel(π_2, π_3) in (17a) forced by the particle *too* and Narration(π_2, π_3) in (17b) inferred from the presence of *then*—and the rhetorical point of the non-veridical conditional. That is, if the truth of K_{π_2} and K_{π_3} are entailed by the veridical relation, then why is the truth of K_{π_2} asserted only relative to some supposition from which it follows only defeasibly in the first sentence? This is not an outright contradiction, but it makes the point of asserting the conditional very unclear—indeed we might say that the discourse is pragmatically incoherent.

In (17a), however, the attachment works. The scope of the non-veridical conditional relation is over both constituents (π_2) and(π_3) and the Parallel relation. Maximize Discourse Coherence will force us in effect (π_2) and(π_3) link together to form a new constituent that becomes the consequent of the conditional. This is a coherent discourse structure and there is no clash between veridical and nonveridical relations. But there must be constraints on when can such constituents be formed, because Maximize Discourse Coherence doesn't allow us to form them in all cases —viz. (17b). Given a situation where we have $R(\alpha, \beta)$ and Rs is non-veridical, we will be able to attach γ to β via some veridical relation R', only if γ also bears R to α. This leaves open, of course, the possibility of course of attaching to some parent of α and β, e.g. some constituent that contains both. In earlier work on larger scale patterns of discourse structure, colleagues and I argued that two constituents could be attached together in a subordinate structure to a third one only if the first two bore the same relations to the third— we called this constraint *Continuing Discourse Patterns*. Here we can motivate a similar principle about veridical and nonveridical relations from Maximize Discourse Coherence. The constraint below, mostly informal, nevertheless exploits the SDRT notation $\langle \tau, \alpha, \beta \rangle$, which represents the attachment of a constituent β to α in the discourse context τ.[3]

- Continuing Discourse Patterns for non-veridical relations:

- Suppose from the discourse context one can defeasibly infer that a non-veridical relation R holds between α and β—i.e. $R(\alpha, \beta)$. Then if

  1. one cannot defeasibly infer from the context $\langle \tau, \alpha, \gamma \rangle$ that $R(\alpha, \gamma)$ and
  2. one cannot defeasibly infer from $\langle \tau, \beta, \gamma \rangle$ and the context that $R'(\beta, \gamma)$, where R' is non-veridical.

- then normally: $\neg\langle \tau, \beta, \gamma \rangle$.

This constraint makes conceptual sense: if you want to have a non-veridical relation have scope over a veridical one, you had better make sure that both terms of the veridical relation are also within the scope of the non-veridical relation. A discourse structure in which this constraint doesn't hold will be far from maximally coherent.

I have stated Continuing Discourse Patterns as a constraint only on attachments involving β. But Maximize Discourse Coherence dictates a similar constraint for any attachments to α where a nonveridical relation R holds of α and β. And it is just such a constraint from which we can derive our observation about parentheticals in conditional contexts—namely, that in attaching a parthentical to the antecedent of a conditional, one must use the Conditional relation. For suppose that one wishes to attach new informaton (labelled by π_2) to an antecedent (labelled say by π_1) of a conditional. In order to satisfy Continuing Discourse Patterns, we must attach π_2 to π_1 with a non-veridical relation or we must be able to infer that π_2 also can bear the Defeasible Consequence relation to whatever pi_1's consequent is. As Defeasible Consequence is the *only* non-veridical relation in monologue unless we consider repairs, it is the only non-veridical relation with which to attach π_2.

But why do parentheticals in conditionals apparently *never* have the scopes predicted by the simple account, as the embedding test shows? Couldn't we in fact have a situation of the following schematic sort: we have a conditional *if* π_1, *then* π_3 and a parenthetical π_2 which is attached to π_1 via some veridical relation R but within the scope of the conditional? Thus, we would have something like the following SDRS:

(18)
$$\begin{array}{|l|}\hline \pi, \pi_3 \\ \hline \pi : \begin{array}{|c|}\hline \pi_1, \pi_2 \\ \hline R(\pi_1, \pi_2) \\ \hline \end{array} \\ \text{Defeasible-Consequence}(\pi, \pi_3) \\ \hline \end{array}$$

Note that this situation could only occur if we can derive Defeasible-Consequence(π_2, π_3). But more is at stake here. This sort of attachment will not, for many sorts of conditionals like counterfactuals and normality conditionals, allow us to derive what was asserted—namely, that $K_{\pi_1} \Rightarrow K_{\pi_3}$. A fundamental principle seems to be that the addition of parenthetical information shouldn't change the underlying assertion. It should only add more information. More precisely,

- **Parentheticals only add to asserted content.**
- Suppose that prior to attachment of parenthetical information α, we have asserted content ϕ. Then after integrating α into the discourse context τ to get a context τ', it must be the case that $\tau' \rightarrow \phi$.

But the attachment of π_2 to π_1 would make the original asserted content unrecoverable. So if we adopt this principle about parenthetically used information, we predict that we cannot attach a parenthetical to the antecedent of a conditional by a relation other than Defeasible Consequence.

Continuing Discourse Patterns also makes sense of those examples in which we have attachment to the second term of a non-veridical relation—e.g., to the consequents of conditionals. examples. In (17a) Continuing Discourse Patterns doesn't fire; in addition to the veridical relation Parallel between (π_2) and (π_3), we can establish given $\langle \tau, \pi_1, \pi_2 \rangle$ a *non-veridical* conditional relation between π_1 and π_3. In (17b), however, we can't make the inference to establish that pi_3 can be related via the conditional relation to π_1. The presence of the adverbial *then* breaks this inference. But it also forces us to infer a veridical discourse relation, Narration, with which we must attach π_3. This means then that both conjuncts of Continuing Discourse Patterns are satisfied and so we cannot attach π_3 to π_2. The reason the discourse is odd is that if we don't attach π_3 to π_2, then we can't find, according to the DRT and SDRT constraints of accessibility and availability of anaphoric antecedents, an antecedent for the pronoun in π_3.

An example similar to (17a) that involves parentheticals is (14). On at least one reading, speakers report that the parenthetical can apparently attach to the consequent. In this example, the underspecified element p resolves to the consequent the conditional and we can then specify R to Commentary and attach the parenthetical to the consequent itself. Importantly, the discourse context suggests the truth of the following conditional:

(19) If we have to go somewhere else to get a drink, then it is unfortunate that we have to go somewhere else to get a drink.

Now we can string these two conditionals together, and by transitivity (which holds defeasibly even for normality conditionals) we can deduce the conditional: if the party is over then it's unfortunate that we have to go somewhere else to get a drink. So we can conclude that the relevant nonveridical Defeasible Consequence relation holds between the antecedent and the parenthetical. Thus, our constraint of Continuing Discourse Patterns won't fire, and we can felicitously attach the parenthetical to the consequent in a maximally coherent discourse.

There are, nevertheless, some apparent counterexamples to the application of Maximize Discourse Coherence that relies on Continuing Discourse Patterns. Consider the following (brought to my attention by Frank Veltman):

(20) a. If, as we have just learned, Kim has made an offer, we don't stand a chance.

b. If, as we now know, Kim has made an offer, we don't stand a chance.

Here the author invites his audience to do a simple modus ponens. But note this isn't really a conflict involving attachment. We might readily attach the parenthetical to the constituent formed by the entire previous sentence while nevertheless identifying the underspecified object argument of the propositional attitude verbs with the antecedent. There is a conflict here between the presupposed status of the complement in the parenthetical when it's attached with a veridical relation and its suppositional status as the antecedent to a conditional in the two examples. But the conflict is eliminated by making the inference to a suppressed conclusion. This type of structure is a compressed way of getting from accepted information to a new and perhaps unwelcome conclusion.

One should note that the examples in (20) rely on the particular parentheticals used. The verbs in those parentheticals are *both factive and epistemic* and this seems key to that sort of attachment pattern. Other parentheticals as in (15) don't allow for that sort of attachment. Because Continuing Discourse Patterns is a *default* constraint on attachment, we can admit such specialized rhetorical patterns without inconsistency in this analysis. Default constraints sometimes are overridden, when more specialized contexts, like these in which factive and epistemic parentheticals are present.

Continuing Discourse Patterns is a constraint on attachment. But there are others, as I intimated earlier. Interestingly, when we compare (14) to (15b), we see the attachment preference is determined by the choice of modal. "unfortunately have to...,"unfortunately must" sound fine, whereas "unfortunately should ..." sounds less good. Perhaps it is that the deontic *should* is not something that can be regretted, though vaguer doxastic or epistemic modalities like *must* can easily be regretted. More to the point, if something is to be regretted within a conditional, then normally one might think it is the consequences, unless those consequences are a matter of conditional obligation. Then one should regret the triggering cause or occasion of those obligations. For parentheticals these observations set up then a preference ordering for attaching parentheticals within conditionals not so unfamiliar from presuppositions: attach as low as is consistent with τ, ϕ maximality. In particular the choice of modal affects where a Commentary can be attached.

4 Why do some parentheticals invert subject verb order?

There seems to be a division between parentheticals that happily undergo subject verb inversion and those that don't in English.

(21) The economy is no longer growing, reports the chief economist for Citycorp.

(22) ? Please leave, beg you I.

Informants report that in some languages like German and French with saying verbs inversion is obligatory. In Spanish and Portugese inversion is the default. In English it seems as though the inversion is largely stylistic. An informal survey of Reuters news articles reveals many parentheticals both in normal and in inverted order. On the other hand, there do seem to be syntactic constraints in German and French. It may even suggest that some elements really are extraposed and thus are distinct from the parentheticals analyzed here; that is, they simply are a syntactic rearrangement of one complex constituent rather than two distinct constituents. The syntactic extraposition cases seem, however, relegated largely to verbs of saying, for it is with these verbs that we see the inversions.

5 Other Scope Escaping Elements: Pragmatic Conditionals

There are some other examples of scope escaping elements that, though syntactically distinct from parentheticals and discourse adverbials have similar semantic and pragmatic properties. These are the so called pragmatic conditionals, studied for instance by Haegeman (1984). I give some examples below.

(23) (i) A: Does anyone know any of the applicants?
(ii) B: I know one, Piell.
(iii) A: How do you spell him?
(iv) B: P-I-E-L-L.
(v) A: OK.
(vi) B: If it's the same man, I haven't read his file yet.

being good means doing nothing you wouldn't want me to do as well as doing nothing we would want to do, if you see what I mean.

b. The fault, if it's a fault, is to be found in the system.

c. The story, if it may so be termed, is weak and loose.

d. Piggies, if you remember *Lord of the Flies*,...

e. If you're hungry there's food in the fridge.

f. If you don't mind the word, he's a bully.

g. (i) A: they thought there was something structurally wrong with it, the rear wall if you remember.
(ii) B: which you had taken down?
(iii) A: Yeah.

The analyses of these examples broadly follows the account already laid out for parentheticals. The antecedents of these conditionals are typically *not* giving propositions upon which the contents of the main clauses are truth conditionally dependent. Rather, like parentheticals, they take a wider scope. Many of these provide assumptions upon which the rhetorical functions of the main assertions depend. For this reason, the conditional antecedent takes scope over a particular rhetorical relation with which the main clause material is attached. Take for instance the first example (23a). B's response to A's question that he knows one of the candidates is an answer to that question, *provided* that the claim in the conditional is correct—i.e., provided that the Piell he knows is the one who applied. Thus, the attachment of the conditional is much higher up in the discourse structure than the material surrounding the conditional. The conditional has scope over the discourse constituent containing the two question answer pairs (23a.i-iv) given earlier in the example. Similarly, we might take the conditional *if you see what I mean* in (23b) as a conditional taking wide scope over a question answer pair relation; the conditional says that if you understand what I'm getting at, then this will be an answer to the background question about what it is to remain faithful (see Haegerman 1984 for discussion). Some of the other examples are less clear—for instance, (23d). In that example, *there's food in the fridge* is a background premise to the unstated conclusion that the addressee can eat some of the food if he's hungry. Once again the attachment of the antecedent depends not on sentential syntax, however, but on discourse coherence considerations.

Haegerman analyzes these examples, along with parentheticals, using relevance theory. The antecedents of these conditionals are understood as facilitating access or otherwise enhancing the relevance of the material in the consequents. Haegerman's relevance theoretic analysis of pragmatic conditionals and parentheticals appears to be at a different level from the one proposed here; it tells us the speaker's goals behind the utterance of these conditionals. But it subscribes largely to the non truth conditional view of these items, which it has been my principal aim to rebut. The SDRT analysis given here shows how parentheticals and pragmatic conditionals can make a truth conditional contribution to the content of the discourse.

6 Conclusion

I have argued for the following. Parentheticals and discourse adverbials attach with discourse relations to the assertions or components thereof in which they are introduced. But their scopes are determined by a variety of intricate factores like Maximize Discourse Coherence and the resolution of conflicts between discourse relations. Parentheticals do not have the simple analysis presupposed in the embedding test, because they are distinct discourse constituents and must be attached via some discourse relation that interacts with the conditional. In this way, my account can explain away the phenomena noted by Wilson with the embedding test without endorsing the claim that parentheticals or adverbials have a "non-truth-conditional" semantics.

Notes

[1]Perhaps other parenthetically used verbs like *fear, suspect* may support this reading but they don't require it.

[2]Commentary is a discourse relation in SDRT. For details see Asher (1993).

[3]In Asher 1993, this constraint was built into the much more complicated SDRS update definition given there. In earlier work also continuing discourse patterns looked like the contrapositive of the constraint here and was a "hard " constraint. The restricted version of Continuing Discourse Patterns here is defeasible and for technical reasons we have to use the slightly more complex contrapositive form, $\neg A > \neg B$, which is not equivalent in nonmonotonic logic to $B > A$.

7 References

N. Asher [1993]. *Reference To Abstract Objects in Discourse*, Kluwer Academic Publishers.

N. Asher and A. Lascarides [1998]. The Semantics and Pragmatics of Presupposition, *Journal of Semantics*, **15**, pp239–299, OUP.

P. Grice [1989]. *Studies in the Ways of Words*, Harvard University Press.

J. Groenendijk and M. Stokhof [1984]. *Studies on the Semantics of Questions and the Pragmatics of Answers*, PhD thesis, Centrale Interfaculteit, Amsterdam.

L. Haegeman [1984]. Interjections and Phrase Structure. *Linguistics* 22. pp.41-49.

L. Haegeman [1991]. Parenthetical Adverbials: The Radical Orphanage Approach. In Chiba et al. (eds), *Aspects of Modern English Linguistics*. Tokyo: Kotakushi. pp.232-254.

J. Hintikka [1974]. *Models for Modalities*, Dordrecht: North Holland Press.

A. Lascarides and N. Asher [1993]. Temporal Interpretation, Discourse Relations and Commonsense Entailment, *Linguistics and Philosophy*, **16**, pp.437–493.

C. Roberts [1986]. *Modal Subordination, Anaphora, and Distributivity*, Ph.D. Thesis, University of Massachusetts at Amherst.

D. Sperber and D. Wilson [1995]. *Relevance*. London: Blackwell's.

D. Wilson [1975]. *Presuppositions and Non-truth-conditional semantics*, Academic Press.

Nicholas Asher
Department of Philosophy,
University of Texas at Austin,
316 Waggener Hall,
Austin,
Texas, 78712,
USA
nasher@bertie.la.utexas.edu

Conversational Cooperation: The Leading Role of Intentions

G. Boella, R. Damiano, L. Lesmo and L. Ardissono
Dipartimento di Informatica - Università di Torino

Abstract

In this paper, we consider the role played by the notion of agent intentions in dialog and, particularly, in conversational cooperation.
We present a model, where intentions arise from goals on the basis of two factors: the advantage of achieving a goal, and the cost of the actions to achieve it.
In the model, the adoption of the intention of cooperating at the conversational level is motivated by the fact that the refusal to do low-cost actions, is usually interpreted as an offense to the requester of the action, and that preventing a partner from becoming offended is one of the goals the speaker considers when taking a decision about what to do.
Finally, we show that the decision to cooperate towards the success of the communication accounts for grounding, requests of repair, repairs to misunderstandings, and other related phenomena.

1 Introduction

We assume that dialog is a goal-directed behavior, where the participants realize the existence of some goals of another agent and decide to cooperate (to a certain extent) with him. However, two different types of goals can be identified: domain goals (e.g. knowing the time, or the place where you can buy some gasoline, or getting your friend's car for this night) and dialog goals (e.g. receiving an answer, or a confirmation that a statement of yours has been understood properly; these goals will later be classified as Linguistic or Conversational). In each turn, a speaker expresses one or more goals of both types and the hearer either accepts to cooperate (i.e. adopts that goal) or does not accept.

As noticed in (Airenti et al., 1993), a participant, even when he does not adopt the domain goals of his partner, typically continues to cooperate at a conversational level, by informing him that he has not the required information or by explicitly refusing to perform the requested actions. So, it seems that two different aspects enter into play: one concerns domain behavior, and leaves the hearer free to adopt the partner's goals, one concerns conversational behavior, and often forces the hearer to cooperate independently from his own goals.

(Traum and Allen, 1994) have challenged intention-based approaches to dialog modeling ((Cohen and Levesque, 1990), (Lambert and Carberry, 1991), (Airenti et al., 1993)) arguing that, in non-cooperative settings (i.e. when the speakers do not already share a goal) these approaches leave unexplained why a speaker should bother adopting the goal of being cooperative both at the conversational level and at the behavioral one. In order to overcome these difficulties, (Traum and Allen, 1994) resort to the idea that speech acts pose obligations on the hearer. Obligations provide the motivation for the hearer to act even in non-cooperative situations.

In order to account for these remarks, we provide, in our model, two separate levels of knowledge for conversational and domain behavior (Ardissono et al., 1998b; Ardissono et al., 1999a; Ardissono et al., 1998a). The conversational level takes into account grounding, refusals, and other phenomena, while the domain level includes plans specifying how an agent can achieve his goals in a given domain. At both levels, however, the agent's behavior is driven by his intentions. So, an agent has to form conversational intentions as well as domain intentions out of a set of goals (either his own goals or a partner's goals), and, under some conditions, he decides to adopt the conversational goals of the partner without committing himself to the partner's domain goals.

It is the process of intention formation that determines an agent's behavior. When an agent decides to act, he must carefully (more or less) balance his interest in attaining the goal against the effort required to attain it and its compatibility with other goals. So, even in non-cooperative situations, the adoption of domain or dialog goals is explained in terms of intentions: social goals like preventing the speaker from becoming offended play a role in the decision process, leading an agent to adopt a goal of a speaker even if they do not already share any other goal.

In the next section we present the basic concepts that underlie the proposed model, and in Section 3, we show how they apply to the notion of politeness in agent interaction. Then, we show that, when conversational cooperation has been set up, the model predicts that agents may adopt different conversational strategies. The fifth section describes an implementation of the model and discusses an example. A Comparison with Related Work and a Conclusion Section close the paper.

2 The Underlying Model

In general, we assume that every agent has a set of goals, each associated with a relevance measure, representing the importance of the goal for the agent. Besides goals, an agent has a set of preferences towards states of affairs, with associated a utility function representing the advantage for the agent of (even partially) achieving a certain state of affairs. A goal may also occur among the preferences of the agent and, hence, it may be related to some utility function for evaluating the advantage of achieving it.

Goals and preferences, however, play different roles: goals are the input to the planning process, while the utility functions expresse the preferences of the agent among the plans which satisfy a certain set of goals. In this way, the advantage of a plan is not only evaluated with respect to the the fact that it achieves the goals the plan has been built for, but also with respect to the advantages provided by the side effects of the plan (for instance, consuming less resources).

So, the "goal situation" of an agent A, i.e. G_A, is defined as a set of couples:

$G_A = \{<g_1, r_1>, <g_2, r_2>, \ldots <g_n, r_n>\}$

where g_i is a conjunction of literals, the relevances r_i are positive real numbers.

The preferences of an agent are expressed as a set of couples as well:

$U_A = \{<s_1, u_1>, <s_2, u_2>, \ldots <s_m, u_m>\}$.

where s_i is a conjunction of literals and u_i is a function from world states (set of literals) to real numbers.

The overall utility function that an agent exploits to evaluate the world state S resulting from the execution of a plan is a weighted sum of the utility functions associated with each preference:

$UF_A(S) = \sum_{i=1}^{m} w_i u_i(S)$

The relevance expresses an a-priori evaluation from agent A about the increase of his well-being in case he decides to adopt a line of behavior aimed at achieving the goal expressed by g_i. Since an agent cannot at any moment build a plan for satisfying all of his goals, he has to focus on a subset of them: the relevance of a goal is the criterium for chosing to plan for satisfying only a subset of them. The set of currently focused goals R_A is composed of those goals in G_A whose relevance is greater than a given threshold r_A (that can be varied according to the resources he can allocate to planning and deciding what to do).

Moreover, each agent knows about a set of action schemas $AS_A = \{AS_1, AS_2, \ldots, AS_k\}$ where each AS_i is a 5-tuple: $AS_i = <Name_i, Prec_i, Constr_i, Eff_i, Decomp_i>$:

1. $Name_i$ belongs to a predefined set of Action symbols, each with an associated arity.

2. $Prec_i$, $Constr_i$, Eff_i are conjunctions of literals (preconditions, constraints, and effects).

3. $Decomp_i$ is a set of sequences $Subact_{il}$, where the elements of $Subact_{il}$ belong to AS_A; each $Subact_{il}$ represents a possible decomposition of AS_i; if $Decomp_i$ is the empty set then AS_i is called Executable, otherwise it is called Complex.

On the basis of his focused goals R_A and of the actions he knows, an agent A can build a set of plans, by selecting the actions which have among their effects one (or more) of the goals in R_A. This is done by means of an algorithm which examines the possible decompositions of the actions. This algorithm builds trees of instantiated actions where, for each node AS_m, the (ordered) sequence of daughters corresponds to one of the decompositions in $Decomp_m$. Each such tree is called a Plan.

A Full Plan is a tree of instantiated actions where all the leaves of the tree are Executable Actions. A plan where some leaves are Complex is a Partial Plan.

According to the definition, an agent is able to obtain an evaluation of the utility of a plan both for Full Plans and for Partial Plans, since the utility is evaluated with respect to the resulting final state: the partial plans just abstract away from some of the side effects resulting from the different possible ways of refining the plan.

When the search algorithm concludes the search either because all plans have been refined or the time allocated to the planning process has been consumed, the agent may choose the plan with the greatest associated utility: he becomes committed to that plan, which constitutes his current intention I_A.

The process of intention formation for single agents can be summarized in the following way:

1. The set R_A is composed of those goals in G_A whose relevance is greater than the threshold r_A.

2. A builds the set of plans P for achieving the set of goals R_A.

3. A choose sthe p_i in P such that its outcome maximizes his utility.

The situation becomes more complex when a group of agent interact. In particular, it often happens that some goal g_B of agent B becomes known to agent A. A special case of this situation arises in case B has explicitly asked A for help. In these cases, it is possible that agent A comes to insert this goal among his intentions even if A has no preference towards this goal. The reason for doing so is the fact that doing something for achieving the goal of someone else can anyway have some utility for A: as stated above, a plan is evaluated also with respect to the side effects it produces.

What's more important is that the whole process of intention formation has now to take into account the presence of another agent. In particular, whenever an agent A has included among its goals a goal originating from another agent B, we say that A is pre-cooperating with B. Notice that, in general, it may happen that A will never do anything to satisfy g_B. However, the fact that A takes into account g_B is a (weak) form of cooperation: at least the agent spends his time and allocates planning to take into consideration the alternative of adopting the partner's goal. In case A chooses a plan that leads to the satisfaction of the partner's goal, we say that A cooperates with B. In case that B is aware of the formation of this intention, then the agents are cooperating to a shared goal (Boella et al., 1999).

Besides considering the possibility of doing something for achieving other agents' goals, in an interaction process, an agent should try to foresee their possible reactions. Not only in case their goals have been adopted and achieved but also in case the agent decides that it is not worth for him to do anything for satisfying those goals.
Only by considering the situation resulting after the reaction of other, the agent can really assess the advantage of a given plan.

A kind of "anticipation feedback" of the reaction of the partner is, therefore, required. (Ndiaye and Jameson, 1996) has adopted this form of reasoning in a dialog system in order to evaluate the goodness of the alternative moves of the system. In (Boella et al., 1999), we have exploited a similar form of reasoning in order to evaluate the goodness of an action during the execution of a shared plan by a group of agents. In this paper, we will exploit the ability of an agent to foresee the possible reactions of another agent in order to decide whether it is worth for him to decide to satisfy the other agent's goal.

If an agent evaluated the utility of a plan that achieves a goal requested by another agent only on the basis of its outcome, he would never choose that plan in a non-cooperative setting (i.e., the agent does not already share any goal or preference with the other agent): achieving it does not produce any utility for him. As stated above, the utility of a plan is provided by summing the utility deriving from the preferences of the agent.
Often, performing an action for achieving another agent's goal results only in a negative utility, since the side effects of the action affects other states as resource consumption. In case of cooperation, instead, the partners share some goal which is preferred by both of them, so that achieving it increases the agent's own utility and the utility of the group's as well.

By evaluating the utility of a plan according to the world state resulting after the other agent's reaction, the agent can evaluate how this reaction affects his preferences, for instance not offending the partner and other social goals.

The planning process of the agent in a situation involving interaction, therefore, must also include considering the alternative of doing something for the other agent when his goals are recognized and evaluating the utility of the world state resulting from the partner's reaction.

Considering a goal of the partner during the planning process means assigning it a sort of relevance since planning is one of the resources of the agent. Moreover, this assignment of relevance must not necessarily result in the intention of performing the goal: the agent chooses to achieve the goal only if he can achieve some more utility in doing it with respect to not doing it.

The intention formation process in case of interaction with other agents is the following one:

1. The set R_A is composed of those goals in G_A whose relevance is greater than the threshold r_A.
2. If A knows that there is a goal g_B of agent B such that *relevance-eval*(g_B) $> r_A$, then he creates set R'_A as the union of $\{g_B\}$ and R_A.
3. A builds the set of plans P as the union of the plans resulting from building plans for achieving the set of goals R_A and R'_A.
4. For each plan p_i in P, A considers the possible reaction of B: the world state resulting from the reaction becomes the outcome of p_i.
5. A chooses the p_i in P whose outcome maximizes his utility.

The assignment of relevance by the function *relevance-eval* can be conditioned to the situation: the agent can have no time to waste for considering if it is useful to cooperate or he can know by experience that some goals can be discharged without consideration.
Note that the behavior of considering the goal of other agents does not derive from some social obligation, but it is just relevant in the sense that an agent thinks he can gain a better utility in doing so.

Within this rather wide framework, this paper focuses on the role of intentions on two rather specific problems, i.e. the adoption of other agents' goals during the conversation.

In particular, we claim that:

1. In interactive settings, the involved goals are of three types: Linguistic goals, Conversational Goals, and Domain goals.

2. Agents consider only a limited number of goals at a time depending on their relevance.

3. Agents choose what to do on the basis of their ability to foresee how a given plan changes the situation (i.e on the basis of the utility of that plan).

4. In a group activity, the evaluation of the utility of a plan must involve also the effects resulting from the reaction of the other members of the group.

5. These effects can be accounted for by trying to foresee the actions that a partner will undertake in response of an agent's action (*Anticipation Feedback*).

3 To Be Polite or not to Be

In this section, we assume that:

1. There is an agent A, who is carrying out some plan on the basis of a current intention. According to the model outlined in the previous section, this means that A has previously chosen a plan P_A, which maximizes a utility function.

2. Another agent B has addressed a request to A of satisfying a goal g_B.

The goal of this section is to explain how A can choose to be polite or impolite by adopting or not the goal of B. In particular, since there is, in principle, no utility for A in case B achieves one of his goals, the aim is to show why, in some (possibly most) cases, A chooses a polite line of behavior.

We have mentioned a "request" addressed by B to A. This should not be viewed as a restriction on the kind of thing that B could have said to A. In fact, we consider a question as a request to provide an agent with some information and a statement as a request to the hearer to update his knowledge base.

As stated above, the model involves three types of goals. In the case of a request they are:

1. Linguistic goals: the goal to understand the propositional content of what B has said, together with the subsidiary goals of hearing the voice (in case of vocal communication) of B, of understanding the words composing the sentence uttered by B and of putting the words together in a meaningful expression.

2. Conversational goals: the goal of maintaining the communication channel open, together with the subsidiary goals of keeping B informed that you have understood (grounding) and of maintaining the coherence of the dialogue.

3. Domain goals: any goal of B concerning the external world extracted from the comprehension of the current utterance or of the sequence of utterances composing the dialogue.

We are not going here to describe how these various goals are kept together in a unified structure: in (Ardissono et al., 1999a) we have shown that the problem solving activity expressed as *Agent Modeling Plans* could constitute the required glue; we just assume that as soon as A hears B, these goals are set up inserting them among the current set of goals of A if they are considered relevant.
Of course, domain goals are taken into account only as soon as they are recognized, i.e. depending on the success of the linguistic goals.

Why is it much more common that an agent refuses to cooperate at the domain level than at the conversational level? We claim that the choice of cooperating does not depend on obligations established by the speech acts (as (Traum and Allen, 1994) claim instead); on the contrary, it depends on the type of the action needed to establish cooperation: during the process of forming intentions out of goals, if the cost of the required action is low, a refusal can be motivated only by a negative attitude towards the requester. Therefore, the requester, which can infer the requestee's reasoning, will be offended by a refusal.

The effect on the requester can be foreseen by the planing agent in the anticipation of his reaction: the agent simulates the reaction of the partner depending on his choice to adopt or not to adopt the requested goal.

This explanation is in line with the notion of politeness as defined in (Brown and Levinson, 1987). They show that politeness can be explained with reference to the notion of face. In (Ardissono et al., 1999b), we have provided a formal explanation of how indirect speech acts prevent the hearer from being offended by referring to the interactants' face wants; moreover, we have described how the type of the requested actions affects the offensiveness of a request; compare, e.g., asking the time vs. requesting some money.

In a similar way, we can explain why refusing cooperation at a conversational level may as well threaten the speaker's face wants: paying attention to people, listening and understanding them are "free goods" (Goffman, 1967), so no one can refuse them without threatening the speaker's face. Indeed, ignoring someone and not responding to his speech act is usually perceived as a very aggressive behavior.

In our model, the social goals are expressed as the preference of not offending the other agents, a preference with associated a certain benefit: this utility is part of the overall utility function of the agent, hence it is used to select plans that do not have offending as a side effect.

Note that offending is not a direct effect of not adopting a goal or of action as refusing cooperation; instead, the state in which the partner feels offended is a result of the reaction of the partner when he is faced with the refusal of a low cost action.

As we showed in the previous section, if the goal is relevant, the agent has to compare the plans that achieve just his private goals, with those satisfying the partner's goal as well. Moreover, he will consider also the reaction of the partner in case the goal has been adopted or not. Hence, he has the information for deciding whether it is better to save his own resources and risk to offend the partner, or to spend some of his efforts and preserve the social relationship with him. The choice of the agent depends on the relative weighting of the utilities deriving from the preferences of saving resources and not offending the partner.

There is another situation, however, in which the other agent's goal are not adopted: when, for some reason, a low relevance is assigned to that goal. This behavior expresses the fact that the agent does not consider worthwhile to allocate planning resources for deciding what to do about the goal.

So, the basic tenets of the model are that:

- the relevance is determined without involving means-ends reasoning, but just on the basis of an a-priori evaluation of goals (possibly based on learned rules);
- the utility is determined involving means-end rationality, and it is what always determines the actual behavior of agents,

  but
- the relevance can be assigned values that prevent the evaluation of utility, thus producing (partially) "irrational" behavior (that is, had the agent have enough resources to evaluate all possible plans, he would have found that doing something for achieving the partner's goal was the optimal solution).

A last point needs to be made clear before going back to the computational model. *Relevance-eval* is a heuristic function: it is not, and should not be, independent of the final utility evaluation. This is rather obvious for single-agent intentions: if the goal g is assigned high relevance, then it means that the agent believes that it affects very positively his well-being. So, in the evaluation of any plan leading to g, a highly positive contribution to the global utility should come from the truth, in the resulting state, of g.

Less obvious is the fact that also in two-agents settings the same must happen (if *Relevance-eval* is a good heuristics). If A is used to take into account in its utility evaluation the positive or negative reactions of his partner(s), then the *Relevance-eval* function he uses must assign a reasonable level of relevance to the partner's goals (otherwise his utility function is of no use). Conversely, if *Relevance-eval* assigns high relevance to other agents' goals, the utility function must take into account the partner's reaction (otherwise, considering the partner's goal involves a waste of resources in computing the utility of plans that will seldom be chosen).

4 Conversational Phenomena Accounted for by the Model

We have discussed in the previous section the motivations for starting up conversational cooperation. Once an agent has decided to cooperate with another agent at the conversational level, a group is formed around the conversational shared goals. In our model, shared goals don't compel agents to act, but, instead, within the context of a group, they drive the evaluation of the utility of alternative actions on the part of the agents, given the probability that the actions result in certain outcomes.

In general, we assume that - homogeneously with other behavioral levels - the conversational behavior of an agent can be modeled by a library of conversational plans including actions that encode the knowledge about speech acts, adjacency pairs, and so on, and also optional grounding actions, like notifications, requests for notifications, repairs to misunderstanding, and so on (see (Ardissono et al., 1999a)). Here, we don't address the problem of how an agent chooses the form or the content of his speech-acts; instead, we focus on the intention generation process. In the case of conversational cooperation, our model prescribes that the agent builds and evaluates two alternative plans: the first of them includes the conversational action under consideration, and we will call it *rich plan* (*RP*); the second one does not include it, and we call it *simple plan* (*SP*).

The presence of the conversational goal in the *RP*s is in some cases explained by the fact that the agent has the goal of checking whether his action succeeded (for instance, the success of a speech act), as well as that, on the other side, the partner considers to adopt this goal when he knows that the other agent wants to achieve it.

These plans are then evaluated in the light of the group's utility, and the *SP* or the *RP* is chosen depending on the result of the evaluation: when evaluating the utility of a plan, the agent takes his partner potential reaction into account.

4.1 The goal of understanding a speech act

As long as the conversation proceeds smoothly, the task of checking the partner's understanding in a conversation is normally accomplished by each participants by exploiting the implicit feedback that the partner's next turn provides. However, participants sometimes detect difficulties in understanding the partner, or in making themselves clear, and, in some circumstances, they act in order to prevent such difficulties, when possible. In all the cases, the advantage that is more or less directly traded off (by means of the utility function) against the effort required to undertake some kind of grounding action is the predictable saving of resources that stems from avoiding, in the future, a clarification or negotiation phase to restore the lost common ground.

- The speaker (A), given his beliefs about the common interpretation of the interaction up to that moment, is normally aware of the difficulties the hearer (B) may find in interpreting his turn. Moreover, he has the goal to know whether B succeeded in his interpretation. So, A can decide to take the initiative and explicitly check B's understanding; in order to do so, A may solicit feedback using discourse markers like "right?", "ok?", etc. Based on these factors (predicted difficulties, importance of the turn, etc.), A evaluates the utility of the *RP* in which he checks the B's interpretation and the utility of the *SP* where he doesn't. In case the *RP* gets a higher utility value, then A will commit to the plan that includes an "ask for notification" action.

- Knowing that A has the goal of checking whether B's action of understanding succeeded, B sometimes adopts this goal and notifies to A the success of the interpretation process ("mhm", "right" and repetitions) or his failure ("what?"). However, since B's next turn usually displays his understanding, these notifications are often by-passed. From the perspective of our model, B has to compare the utility of the *RP* including the action "Ground" with the utility of the *SP* consisting in going on with the new turn without any notification.

- Finally, A often tries to facilitate B in his understanding process. In a coherent dialog, each new turn is usually related in some way to the current dialog focus (Carberry, 1990). In general, topic shifts are costly for B because in order to identify them, he has to search for a link within the context starting from the current focus. Moreover, the clearer A's turn, the higher the probability that B understands it correctly, with a valuable saving of resources for the group cooperating at the conversational level. A's *RP* consists in a plan where he exploits the linguistic resources at his disposal to make the focus shift apparent, while the *SP* is one where he simply changes the focus without explicitly signaling it to B. If *RP* yields a higher utility, because B can interpret the sentence with less effort, A will introduce cue phrases like "however", or will notify that a new topic is introduced ("by the way").

4.2 The goal of maintaining a common interpretation

For the dialog to proceed coherently, the interactants need to share the same interpretation of the previous part of the interaction, or better, the mutual belief must hold that their respective interpretations are reasonably aligned.[1]

The lack of interpretation problems is, by itself, a symptom that there is a common interpretation, i.e., that participants' interpretations are reasonably aligned. Of course, this does not mean that the interpretation is really the same, but only that the possible differences fall within the standard individual differences (Ardissono et al., 1998a). In absence of specific signs of misalignment, this normally makes the utility of any active effort to check the interpretations' alignment very low, because the effort required would not be compensated by the low risk of having to initiate a negotiation phase in the following, to restore the lost alignment.

On the contrary, the loss of dialog coherence normally means that a misunderstanding has occurred, i.e. that at least one of the two speakers has chosen a wrong interpretation of a turn (Ardissono et al., 1998a). If a participant A realizes that a misunderstanding has occurred, he will form and compare two alternative plans: the *RP* includes a appropriate action for addressing and solving the misunderstanding, the *SP* has no extra action to address misunderstanding and consists in simply going on with the next turn. If the comparison between the two alternatives ends in favor of the former one, the participant will act in order to find out who is the misinterpreting agent, and correct the wrong interpretation. Intuitively, the utility function here embodies the idea that a misunderstanding is addressed if it is deemed relevant, and is at risk of posing difficulties for the subsequent interaction.

1. if A realizes that his B is the misinterpreting agent, the *RP* will include a request for repair ("No, I mean that..."); B, in turn, is expected to repair his

interpretation as requested, and notify the execution of the repair to A ("Oh, ok ...").

2. if A realizes that he has misinterpreted B, his *RP* will include the repair and the notification to B that now he holds the right interpretation ("Oh, you meant that ...").

4.3 Other conversational phenomena

In our model, a number of other phenomena occurring in a conversational context stem from behavioral cooperation, i.e. once it has been established the shared goal of performing a (linguistic or domain) action requested by the partner:

- The hearer's notification that it is impossible to perform the requested action:

  A: Could you tell me the time, please?
  B: I'm sorry, I don't have my watch on me.

  In this example the *RP* includes the notification that the requested action is impossibile to perform. These notifications are captured in our model of cooperation (Boella et al., 1999) by the fact that informing the partner that requested action cannot be performed usually results in a higher group utility, because it prevents the other group members from wasting further efforts in vain (for instance, by repeating the request).

- Conversely, B often notifies that he has done the requested action:

  A: Could you press the key for me?
  B: Done!

- When B has adopted the speaker's perlocutionary goal that he (B) intends to update a certain belief, B sometimes notifies that he believes what has been communicated, possibly after a solicitation by A:

  A: we have a boxcar of oranges in Amsterdam, right?
  B: right

  Here, A's choice to include in his plan the solicitation of a feedback is motivated by his goal to check that B, after recognizing his communicative goal (the mutual belief that the A wants B to intend to adopt the belief conveyed by the assertion), actually tries and updates his "knowledge base" with this belief.[2]
  B knows the partner's goal, so, he can consider to adopt it even if he is not solicited: providing spontaneously the notification results in a less costly dialog in that A has not to check explicitly whether B succeeded.

- Negative notifications of update are motivated by the same goal:

  A: obligations do not exist
  B: I don't think so

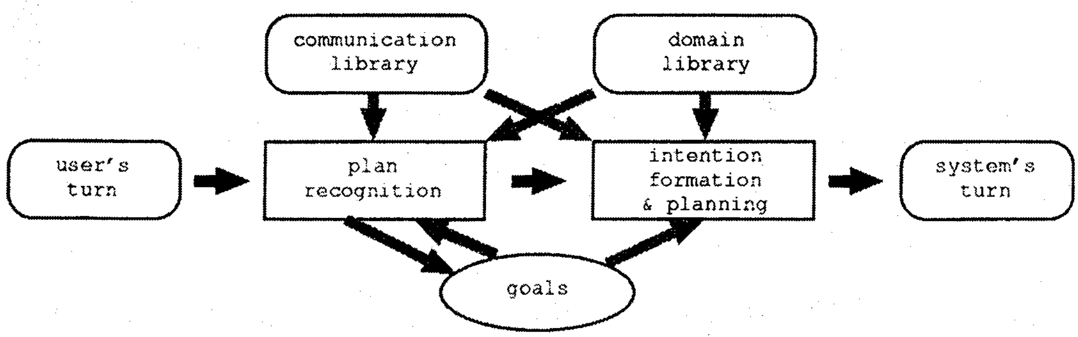


Figure 1: The architecture of the dialog system.

5 The Agent Architecture

The dialog component of our system is described in Figure 1. Although a separation is shown between the plans for achieving Linguistic and Conversational goals (Communication Library) plans and the plans concerning domain actions (Domain Library), the internal organization of the two libraries is exactly the same, the main difference being that the communication library is domain-independent. Both libraries are controlled by a reactive planner, which selects the actions that the agent will do next; the selected action constitutes his intention.

As stated in Section 2, the system (S) applies a planning process driven by the goals and the utilities associated with possible courses of actions. When S realizes that another agent (U) wants S to adopt a new goal by means of the plan-recognition process (Ardissono et al., 1999a), S applies *Relevance-eval* in order to get the relevance value to associate with that goal. If the relevance is greater than the current threshold r_S, then S inserts the goal in the set of the goals to consider, in order to see whether it is worthwhile to adopt it.

The box in Figure 1 labeled "intention formation & planning" is expanded in Figure 2 in order to show more precisely how the utility of a plan is evaluated.

S, given his goals and the one identified during the plan recognition phase, selects some applicable plans and considers the possible outcomes deriving from their execution (S_1, ... S_n). If he limited himself to the outcome of his own actions, when evaluating the alternative plans, he would never adopt any goal from U, since this would always result in a waste of resources, without any other immediate benefit.

Therefore, S has to consider also the potential reactions of U, given U's goal that S performs the action U requested (see the connection in Figure 2 between the recognized goal and the simulation of the planning of S): this is done by means of another instance of the agent architecture which emploies U's point of view. The simulation of U's reaction results in a new set of outcomes: only by considering the utility of these outcomes, S can assess the advantage of his choice; in fact, the reaction of U may affect some of the social goals of the system (for instance, the desire not to offend the partner). Intuitively, not adopting one of U's goals may result in U being in a bad-tempered mood, a fact that could have serious consequences for S. So, the adoption of the goal prevents the offense of the partner, and a social resource is saved at the expense of some effort for executing the action. A similar algorithm, but applied to cooperative settings, has been described in (Boella et al., 1999).

In order to verify the feasibility of exploiting social goals for motivating cooperation, we have implemented a prototype using the decision theoretic planner DRIPS (Haddawy and Hanks, 1998). DRIPS exploits hierarchical plans to find the optimal sequence of actions under uncertainty, based on a utility function. Goals can be partially satisfied, by trading them off against cost (waste of resources) and against each other.[3]

In the experiments carried out to validate the model, we have introduced four different attributes to depict the situation in which S is interrupted by U, while he is executing an action *Act* formed by two steps $Step_1$ and $Step_2$:

- *time*: it models the time as a bounded resource. The utility gets a fixed value in case its two steps are completed before time t_{end}, otherwise, the utility decreases more or less sharply when the action is completed after the deadline as a function of *time*;
- *ground*: it models the U's goal of knowing that S has successfully interpreted the request; when it is true, U infers that S is cooperating at the conversational level.
- *goal*: the action requested by U has been successfully executed;
- *res*: it models the use of (generic) resources. The utility decreases when the value of *res* decreases;
- *offended*: it models the (foreseen) degree of offense of U. The utility decreases when the value of *offended* increases;

Moreover, we have introduced different actions, the most important of which are:

- Action *Ground*: it makes U know that his request has been successfully interpreted.
- Action *Adopt*: it represents S's cooperation with U at the domain level; the decrease of available time and of generic resources depends on the particular action needed to satisfy U's request.
- Action *Refuse*: it represents S's act of communicating to U that he does not intend to do what U requested of him. Among its effects, there is the fact that U comes to know about S's decision (*refused*) and that *ground* is true.

Other actions concern the possibility of providing U with extra help (by having recognized some of his implicit goals).

On the side of U, we have introduced the single action *Evaluate* (see Figure 4), modeling the change of the *offended* parameter, depending on S's action. Notice that

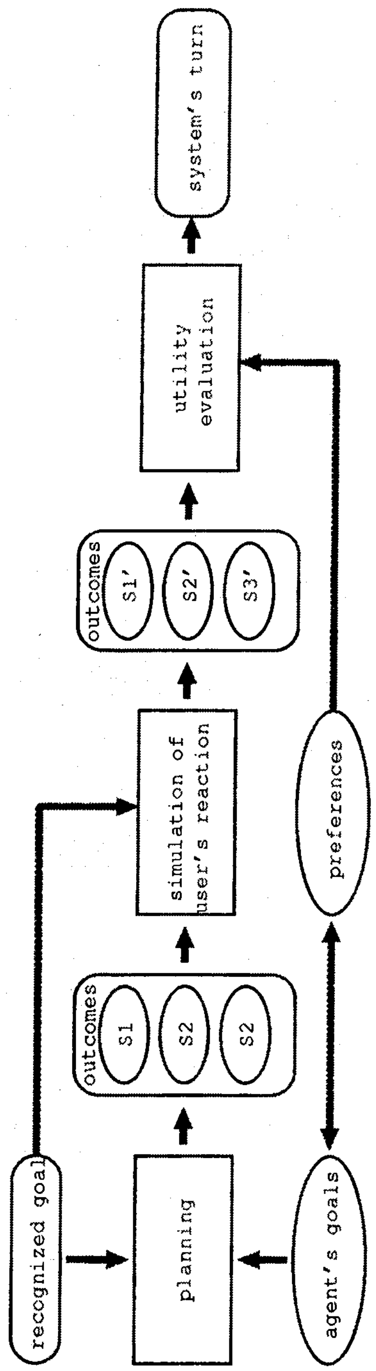


Figure 2: The architecture of the planning module.

```
(action Ground
        (time = time + 1)
        (res = res - 1)
        (ground = 1))

(action Adopt
        (time = time + length(action))
        (res = res - cost(action))
        (ground = 1)
        (goal = 1))

(action Refuse
        (time = time + 5)
        (res = res - 5)
        (ground = 1)
        (refused = 1))
```

Figure 3: Some of the system's actions. The *ground* parameter represents the fact that the goal of grounding the new turn has been accomplished.

what *Evaluate* actually represents is how S believes U will react to his choice. The key parameter affecting the level of offense is the cost of the requested actions: the less the cost of the requested action, the greater the offense; this follows the principle by (Goffman, 1967) that people get offended when they are refused low-cost help.

The *goal* predicate represents the fact that the effect of the requested action has been achieved. The lack of grounding is interpreted by U as S is not cooperating at the conversational level: since cooperating at the conversational level (interpreting the sentence, grounding it) has a low cost, it is offensive not to do it. Note that, besides the *Ground* action, also the adoption of the domain goal or a refusal show S's cooperation at the conversational level. On the other hand, in case of domain action there are different possible costs and also differences between the results of cost evaluation by the two interactants (see the parameter *action*). The *higher-goal* parameter in *Evaluate* is set to 1 as an effect of the action *Extra-Help* mentioned above.

Now let's consider in detail the current situation, i.e, the one where U has just asked to S to do something while S was going to perform the first step of *Act*. Assume that the heuristic function of S (*Relevance-eval*) has returned a value such that U's recognized goal is considered during the subsequent planning phase. In order to explore the different alternatives, the planner builds and evaluates some plans. These plans differ in the actions for pursuing the partner's recognized goal: they can be included or omitted. It then tries to foresee the reaction of U, and commits to the plan which yields the greater utility according to S's preferences.
The experiments have shown how a change in some parameters affects the behavior of the system:

- By associating a greater utility to the completion of *Act* than to *offended*, and by keeping t_{end} tight (i.e. t_{end} coincides with the sum of the durations of $Step_1$ and $Step_2$) we obtained that S decided not to cooperate at the domain level. The

```
(action Evaluate
       (time = time + 1)
       (res = res - 1)
       (offended = offended + (20 * not(ground)) +
                              (20 * not(goal) / cost(action)))) -
                              (20 * higher-goal))
```

Figure 4: The partner's reaction.

Ground action was executed depending on how fast the utility of *Act* goes down to zero after the deadline and on the utility of not offending the partner.

- By relaxing t_{end}, so that S has time enough to answer to U and complete *Act* within the time constraint, we obtain a fully cooperative behavior: the sequence $<$*Adopt*, $Step_1$, $Step_2$$>$ was the preferred new plan (the requested action is not expensive and the grounding is assured by adoption);
- By increasing the utility of the resource *res* with respect to *offended*, no adoption was produced, or at best the planner explicitly refused behavioral cooperation: the agent was more worried about not wasting his efforts than about preventing U from being offended.

6 Related Work

In order to overcome the difficulties of existing intention-based approaches, (Traum and Allen, 1994) defined an interaction model based on the notion of obligation. Obligations are pro-attitudes that lead humans to act but impose less commitment than intentions; their social character explains why humans are solicited to act. (Poesio and Traum, 1998) provide an axiomatization of speech acts where communicative actions have as an effect the imposition of obligations on the two interactants.

Our model has some advantages with respect to obligation-based approaches. First of all, the same phenomena are accounted for without introducing further propositional attitudes. Moreover, the notion of conversational cooperation together with the definition of shared goal (Boella et al., 1999) provides an explicative treatment of these different phenomena that is uniform with the rest of the model.
(Traum and Allen, 1994) claim that discourse obligations are learned social norms; on the contrary, in our model, the motivation for adopting a conversational goal is given by the goal of preventing a potential offense for the speaker together with its potential consequences. All the speaker has to learn is under what conditions humans happen to be offended, and this same knowledge explains the use of indirect speech acts (see (Ardissono et al., 1999b)).

In (Traum and Hinkelman, 1992) speech acts are modeled as "multiagent actions [...] that are not completed unless/until they are grounded". Instead, in our model, speech acts are single agent actions; similar to domain actions, they succeed only if their effects hold, so agents check in the world looking whether the effects hold. There is no clue that linguistic actions should be modeled in a different way from domain ones.

The only difference is due to the fact that the effect produced by a linguistic action is sometime difficult to detect: the effect of an illocutionary act is to make the speaker's communicative intentions mutually believed. In some cases, the only viable way the speaker has at his disposal to understand whether the linguistic action succeeded is some form of explicit grounding by the hearer.
Finally, the ordering among the alternative sources of deliberation is provided by the means-ends relation involving the different linguistic actions, instead to be explicitly stated as in the model of (Traum and Allen, 1994); for example, explicit grounding actions often are avoided in presence of answering: the explanation relies on the fact that, if the hearer has answered coherently, then the goal of showing his understanding of the question has been achieved and an explicit grounding action is superfluous and results in a further consumption of resources.

The strategy of anticipating the partner's potential reaction, when choosing the next action to be performed in a conversational context, has been exploited also by (Ndiaye and Jameson, 1996) in the dialog system Pracma. In this work, the authors propose a technique called "global anticipation feedback" to select the next system's move, in association with an evaluation of the utility of the available moves; this technique consists in the system temporarily assuming the role of the user, in order to make an assessment of the potential user's reaction to a given system's move. Once the user's reaction has been generated, for each alternative move available to the system, the utility of both system's and user's move is evaluated, and the move that maximizes the sum of the two is chosen. However, in both cases the utility esteem is made from the point of view of the system; moreover, utility is an attribute of the move, i.e., is not evaluated on the basis of explicit goals motivating the system's behavior.

In order to make the system's choices more principled at a higher level, an obligation based approach has been introduced in (Jameson and Weis, 1996), based on (Traum and Allen, 1994). In this model, obligations can be set by certain move types, during the course of interaction, and their status, ranging from acceptance to rejection, can be subsequently affected by other moves. Every move has some methods associated to it, that allows to specify what obligations it creates or addresses. Obligations, in turn, have some properties like a penalty that is imposed to the system for rejecting the obligation itself, or, possibly, an expiration period. For each obligation, the moves that address can be specified, together with the corresponding changes in the status of the obligation. This knowledge is exploited to constrain the system's planning of the next move, in order to imitate the way human participants address obligations in real interactions, especially in non cooperative settings. However, the obligations considered here are domain-dependent, since they are tailored to a particular dialog type, and they utility with respect to the system's goals is accounted for only in an implicit way. The utility of a move is then evaluated with reference to how it affects discourse obligations, both existing and new ones. In order to show the improvement resulting from taking obligations into account, the authors depict different behavior types, from the simplest, where obligations are not considered, to the most complex, where the system accounts for the obligations of both speaker and hearer. However, no anticipation of the moves of the speaker is used in the latter hypothesis.

Although anticipation feedback and dialog obligations can be easily integrated in the same system, none of the two techniques constitutes, by itself, a general and uniform explanation of the conversational behavior of the system.

7 Conclusions

In this paper we proposed an intention-based approach to dialog that aims to overcome the critics posed by (Traum and Allen, 1994). Our solution does not employ the notion of obligation, even if some similarities can be found with (Traum and Allen, 1994), since they use obligation for explaining why agents adopt goals of other agents, even in non-cooperative settings: the reason for not exploiting the notion of obligation is to have a uniform source of motivations for explaining the behavior of agents.

Some considerations on possible critics must be done. First, reducing the number of propositional attitudes, the reasoning process becomes more complex; our model wants to be an explanation and it does not exclude the possibility of compiling the reasoning in more compact form: as (Brown and Levinson, 1987) notice for what it concerns their model of politeness, "there is a rational basis for convention".

Second, here we do not deny the existence of obligations in general: the social character of some dialog behaviors which in (Traum and Allen, 1994) are explained as obligations is accounted for by the reference to social goals (for instance, not offending the partner).

Notes

[1]It can be argued that speakers don't pursue the achievement of mutual belief that common understanding was unachievable, as a consequence of the shared goal to reach mutual understanding. However, when a misunderstanding is not repairable, notifications of the breakdown actually occur (Ardissono et al., 1998a).

[2]Note that the same phenomenon is caused by the shared goal of executing domain actions; e.g. consider "Save the file ... Have you done it?".

[3]In the following example, the DRIPS capability of managing with probabilistic effects and expected utility is not exploited for the sake of brevity.

References

Airenti, G., Bara, B., and Colombetti, M. (1993). Conversational and behavior games in the pragmatics of discourse. *Cognitive Science*, 17:197–256.

Ardissono, L., Boella, G., and Damiano, R. (1998a). A plan-based model of misunderstandings in cooperative dialogue. *International Journal of Human-Computer Studies*, 48:649–679.

Ardissono, L., Boella, G., and Lesmo, L. (1998b). An agent architecture for NL dialog modeling. In *Proc. Second Workshop on Human-Computer Conversation*, Bellagio, Italy.

Ardissono, L., Boella, G., and Lesmo, L. (1999a). Plan based agent architecture for interpreting natural language dialogue. *International Journal of Human-Computer Studies*, to appear.

Ardissono, L., Boella, G., and Lesmo, L. (1999b). The role of social goals in planning polite speech acts. Workshop at 7th Int. Conf. on User Modeling.

Boella, G., Damiano, R., and Lesmo, L. (1999). Cooperating to the group's utility. In *Proc. of ATAL Workshop*, Orlando, FL.

Brown, P. and Levinson, S. C. (1987). *Politeness: some universals on language usage.* Cambridge University Press, Cambridge.

Carberry, S. (1990). *Plan Recognition in Natural Language Dialogue.* MIT Press.

Cohen, P. and Levesque, H. (1990). Rational interaction as the basis for communication. In Cohen, P., Morgan, J., and Pollack, M., editor, *Intentions in communication*, pages 221–255. MIT Press.

Goffman, E. (1967). *Interaction Ritual.* Penguin, Harmondsworth.

Haddawy, P. and Hanks, S. (1998). Utility models for goal-directed, decision-theoretic planners. *Computational Intelligence*, 14:392–429.

Jameson, A. and Weis, T. (1996). How to juggle discourse obligations. In *Proc. of the Symposium on Conceptual and Semantic Knowledge in Language Generation*, pages 259–267, Heidelberg.

Lambert, L. and Carberry, S. (1991). A tripartite plan-based model of dialogue. In *Proc. 29th Annual Meeting of ACL*, pages 47–54, Berkeley, CA.

Ndiaye, A. and Jameson, A. (1996). Predictive role taking in dialog: global anticipation feedback based on transmutability. In *Proc. 5th Int. Conf. on User Modeling*, pages 137–144, Kailua-Kona, Hawaii.

Poesio, M. and Traum, D. (1998). Towards an axiomatization of dialogue acts. In *Proc. of 13th Twente Workshop on Language Technology*, pages 207–222, Enschede.

Traum, D. and Allen, J. (1994). Discourse obligations in dialogue processing. In *Proc. 32nd Annual Meeting of ACL*, pages 1–8, Las Cruces, New Mexico.

Traum, D. and Hinkelman, E. (1992). Conversation acts in task-oriented spoken dialogue. *Computational Intelligence*, 8(3):575–599.

Acknowledgements

We would like to thank Massimo Poesio for the interesting discussion and his suggestions that made this work possible and the anonymous referees that let us improve the final version of this paper.

Authors

Guido Boella, Rossana Damiano, Leonardo Lesmo and Liliana Ardissono
Dipartimento di Informatica - Università di Torino
C.so Svizzera 185, 10149 Torino ITALY
tel. +39-011-6706711
fax. +39-011-751603
email: {guido,rossana,lesmo,liliana}@di.unito.it

Dialogue Acts, Synchronising Units and Anaphora Resolution

Miriam Eckert & Michael Strube
University of Pennsylvania

Abstract

In this paper, we present a method for classifying and resolving anaphora in spoken dialogue. In our model, which is based on Strube (1998), the domain from which potential antecedents for both individual and discourse-deictic anaphors can be elicited is defined in terms of dialogue acts. Our dialogue segmentation method is influenced by Clark & Schaefer (1989), and Carletta et al. (1997) and shows a high degree of inter-coder reliability.

1 Introduction

In this paper, we present an algorithm for the resolution of pronouns and demonstratives in spontaneous spoken dialogues. Most anaphora resolution algorithms are designed to deal with the predominant type of anaphoric reference found in written texts, which involves the co-indexing relations between anaphors and NP-antecedents. In spoken language, by contrast, it is frequently impossible to identify NPs that the pronouns and demonstratives could refer to, meaning that the types of anaphors and their distribution is different in this domain. In addition, the structure of dialogues is less clear than the structure of written texts, with the lack of punctuation, paragraphs and syntactically complete clauses making it difficult to formally define the domain for potential antecedents. For these reasons, applying existing anaphora resolution algorithms to dialogues would result in a poor performance.

In this paper, we present a major extension of the anaphora resolution algorithm described in Strube (1998). This algorithm consists of an ordered list of salient discourse entities (S-List), which provides preferences for the antecedents of pronouns. The main characteristic of this algorithm is that preferences for intra- and intersentential pronouns are dealt with in a unified manner. The update of the S-List and the anaphora resolution are performed incrementally.

Essential to the success of the algorithm presented in this paper is the interaction between the identification and resolution of different types of anaphors and the determination of the domain of possible antecedents. In this paper we describe how we use dialogue act units to provide the structure necessary for the determination of the antecedent domain and also how the algorithm uses them to function as antecedents for discourse-deictic anaphors.

Section 2 gives an overview of our classification system for the different types of pronouns and demonstratives we identified in spoken dialogue. Section 3 describes how we define the domain of possible antecedents for the anaphors. Our resolution algorithm is presented in Section 4 and is evaluated in Section 5.

2 Anaphora in Dialogues

2.1 Individual Anaphors

In the spoken language corpus we examined (the Switchboard corpus, a collection of telephone conversations between strangers about a given topic, see LDC (1993)), anaphors with NP antecedents constitute only 45.1% of all anaphoric references. This number includes all demonstratives and all instances of *he, she, it* and *they* with NP antecedents.

2.2 Reference to Abstract Objects

We classified 22.6% of all anaphors in the corpus *discourse-deictic*. The referents of discourse-deictic anaphors are not individual, concrete entities but abstract objects, such as events, states, event concepts, facts and propositions. Their antecedents are VPs, clauses or sequences of clauses. In Example (1) the demonstrative refers to the event concept referent of the preceding VP. In (2), the demonstrative refers to the proposition expressed by the preceding main clause, and the pronoun *it* refers to the state expressed by the clause *I don't trust them.*

(1) Now why didn't she **[take him over there with her]**$_i$? No, she didn't do **that**$_i$. (sw4877)

(2) A: ...**[we never know what they're thinking]**$_i$.
B: **That**$_i$'s right. **[I don't trust them]**$_j$, maybe I guess **it**$_j$'s because of what happened over there with their own people, how they threw them out of power... (sw3241)

Whilst there have been attempts to classify abstract objects and describe the rules governing anaphoric reference to them (Webber, 1991; Asher, 1993; Dahl & Hellman, 1995), there have been no empirical studies using actual resolution algorithms. However, there are some important characteristics of abstract object reference that research in theoretical linguistics has mapped out and that we make use of in our algorithm (Eckert & Strube, 1999):

Referent Coercion. Instead of assuming that all levels of abstract objects are introduced to the discourse model by the clause that makes them available, it has been suggested that anaphoric discourse-deictic reference involves referent *coercion* (Webber, 1991), that is, the anaphor itself creates a new referent in the discourse model (cf. also Lewis's notion of *accomodation* (1979)).

Preference for Demonstratives. Discourse-deictic reference, as opposed to individual anaphoric reference, is often established by demonstratives rather than pronouns. This is in line with the coercion assumption because generally demonstratives are used for entities which are less salient than those referred to by pronouns (see Gundel et al. (1993)).

Right Frontier Rule. Webber (1991) notes that only text sections which are on the right frontier of the discourse structure tree are available for discourse-deictic reference, i.e., those which are adjacent to the anaphor.

Predicate Information. The type of abstract object is determined by the predicative context of the anaphor (Asher, 1993). For example, a discourse-deictic anaphor in the object position of the verb *do* must refer to an event concept (example 1 above), whereas one in the subject position of the predicate *is right* refers to a proposition (example 2 above).

2.3 Vague Anaphors

A further 13.2% of the anaphors we classify as *vague*, in the sense that the speaker does not refer to a specific linguistic object. The entities referred to by vague pronouns are similar in nature to the discourse-deictic entities because they are also abstract. However, these pronouns do not refer to the referent of a sentence or VP but to the general discourse topic, as shown in example (3).

(3) B.29 I mean, the baby is like seventeen months and she just screams.
A.30 Uh-huh.
B.31 Well even if she knows that they're fixing to get ready to go over there. They're not even there yet –
A.32 Uh-huh.
B.33 you know.
A.34 Yeah. **It**'s hard. (sw4877)

The pronoun in A.34 is not referring to the specific incidence described by speaker B, but rather to the topic of childcare in general. With these pronouns it is impossible to identify a linguistic string in the context that the pronoun is co-specifying with. An algorithm which relies on linguistic surface form can therefore not resolve them and it is important that they are identified.

2.4 Inferrable-Evoked Pronouns

The remaining 19.1% of anaphors constitute a particular usage of the plural pronoun *they* where it indirectly co-specifies with a singular NP, e.g.,

(4) A.20 ... in **the Soviet Union, they** spent more money on, um, what do you call, um, military power than anything. (sw3241)

In this example, the singular NP *the Soviet Union* has the inferrable *inhabitants/population* associated with it. The highlighted pronoun refers to the inferrable despite this not having been mentioned explicitly. We call these *Inferrable-Evoked Pronouns - IEP's*.

It is usually the case that the NP in question refers to a country, a school, a hospital or some other kind of institution. The pronoun then refers to the authority or the population/members of the institution. For this reason they have elsewhere been termed *corporate* pronouns (Jaeggli, 1986; Belletti & Rizzi, 1988).

Our group of IEP's also includes cases where there is no explicitly mentioned institution, e.g.,

(5) A.19 **They** had an interview with ... The general. Stormin Norman...
A.21 Anyway, at the end of it, **they** rolled all of the U S names of the U S casualties – (sw2403)

The plural pronouns in A.19 and A.20 refer to the television authorities without the institution itself having been mentioned. It seems that certain institutions are salient enough that they require no explicit mention.

Both IEP's and vague pronouns are classified as such by default when the algorithm fails to find a compatible antecedent in the domain specified by the dialogue acts (see Section 3).

2.5 Unmarked Anaphors

We do not mark non-referring pronouns and demonstratives such as expletives, subjects of weather verbs and subjects of raising verbs. Also, we ignore first and second person pronouns as these do not pose interesting resolution problems. More difficult to categorise are the pronouns referred to by Postal & Pullum (1988) as *subcategorised expletives*, which they define as being non-referring pronouns in argument positions, e.g.,

(6) When **it** comes to trucks, though, I would probably think to go American. (sw2326)

To identify non-referring pronouns reliably, we use the criterion of possible question formation. In general, wh-questions cannot be formed on non-referring pronouns, e.g., **When what comes to trucks?*

3 Building Synchronising Units from Dialogue Acts

3.1 Dialogue Act Theories

Byron & Stent (1998) point out that it is difficult to determine the center of attention in multi-party discourse because the participants may not be focussing on the same entity at a given point. Our hypothesis is that the attentional state of the discourse participants can be determined with the help of dialogue acts. We assume that acknowledgments are used by speakers to indicate that common ground is achieved and can therefore indicate which entities have been entered into the joint discourse model. Dialogue acts are also important for a second reason, namely that they can be used as the unit to specify the domain for potential antecedents.

There are many theories of dialogue acts and we here briefly discuss those relevant to our own model here. Our common ground assumptions are based on Clark & Schaefer's (1989) theory of contributions (see also Traum's (1994) *Discourse Units* and Nakatani & Traum's (1999) *Common Ground Units*). In Clark & Schaefer's model, each dialogue act is labelled as a *Presentation* or an *Acceptance*. A Presentation and an Acceptance jointly form a *Contribution*. However, Clark & Schaefer's dialogue act labels are also used for larger units. Their rules are recursive and an Acceptance itself can consist of Contributions, for example when a subdialogue is used for clarification purposes. This feature allows discourse structure to be represented. A further important feature of their model is that a single dialogue act may fulfill multiple functions: it can be both an Acceptance of a preceding Presentation and be a Presentation itself.

Carletta et al. (1997) present a more fine grained approach to dialogue acts in their model, which consists of three tiers describing *Moves* (dialogue acts), *Games* (dialogue act sequences) and *Transactions* (subdialogues). Moves are divided into three subtypes – *Initiations*, *Responses* and *Preparations* – and, again, there are are numerous subtypes within each of these to capture a variety of different functions.

We wanted our model to fulfill two criteria: (1) it should reflect the achievement of common ground, and (2) it should be simple enough to allow a high degree of inter-coder reliability. To achieve the first, we use pairs of dialogue acts to form *Synchronising Units*, similar but not identical to *Common Ground Units* and *Contributions*. To achieve the second, we simplify Carletta et al.'s model, ignoring the subtypes and using only an *Initiation/Response*-type of distinction. Furthermore, we did not allow for recursive discourse structure, as presented in Clark & Schaefer's model.

3.2 Dialogue Acts: Units and Categories in Our Model

We assume that the establishment of common ground is indicated by dialogue acts and affects the operations for adding and removing discourse entities from the representation of the attentional state – in our model the list of salient discourse entities (S-list, see Strube (1998)). We divide each dialogue into short, clearly defined dialogue acts – *Initiations* **I** and *Acknowledgments* **A** – based on the top of the hierarchy given in Carletta et al. (1997).

As pointed out in Byron & Stent (1998), the determination of utterance boundaries is difficult in spoken language, as annotators must use criteria which do not depend on punctuation. For this reason we define our units syntactically: each main clause plus any subordinated clauses counts as a separate **I**. Elliptical utterances, which occur frequently in spoken language, are also counted as separate **I**'s even if they are not full clauses.

A's do not convey semantic content but have a pragmatic function (e.g. backchannel). This unit label allows us to make use of utterances with no discourse entities, e.g. *Uh-huh; yeah; right.* Whilst Byron & Stent (1998) and Walker (1998) assign no importance to such utterances in their models, in our model, these constitute a specific type of dialogue act which is used to acknowledge a preceding utterance.

Finally, there are utterances which function as an **A** but also have semantic content, for example answers to wh-questions – this type is labelled as **A/I**. The double label is reminiscent of Clark & Schaefer's model (see previous section), in which a single utterance can fulfill two functions. Expressed in the terms of the dialogue act markup model DAMSL (Allen & Core, 1997; Zollo & Core, 1999), **I**'s are forward-looking, **A**'s are backward-looking and **A/I**'s are both forward- and backward-looking. Only forward-looking dialogue acts require a further response or acknowledgment. Table 1 shows the labelling guidelines from our manual.

Label	Unit Description	Further Acknowledgment required?
Initiation (I)	Statement Question	Yes (if at turn transition)
Acknowledgment/ Initiation (A/I)	Statement following an I Question following an I Answer to a wh-question Answer to a yes/no-question	
Acknowledgment (A)	Vocal signal indicating understanding Word/Phrase indicating understanding Word/Phrase relating previous I to following I	No

Table 1: Guidelines for Labelling Dialogue Acts

3.3 Achieving Common Ground

In order to adequately represent the joint discourse model, we require a further unit which indicates when common ground is achieved. In our model, a single **I** and an **A** jointly form a *Synchronising Unit (SU)*. Examples of this can be seen in Figure 1. Single **I**'s in longer turns (A.81) constitute **SU**'s by themselves and do not require explicit acknowledgment. The assumption is that by letting the speaker continue, the hearer implicitly acknowledges the utterance. In this sense, **SU**'s differ from *Common Ground Units* or *Discourse Units*, which require a response from the other participant to be completed. In our model, it is only in the context of turn-taking that **I**'s and **A**'s are paired up.

The **SU**'s have two functions. Firstly, they are used to indicate at which point the S-list is cleaned up – after each **SU**, discourse entities not referred to again are removed from the list. The second point is crucial to our hypothesis that common ground has an influence on attentional state: we assume that at turn transitions only acknowledged **I**'s become part of an **SU**. If at a turn transition one speaker's **I** is not acknowledged by the other participant it cannot be included in an **SU** and its discourse entities are deleted from the S-List.

An example of this is given below in Figure 1. In turn B.84, the entity *our area* is added to the S-List. However, Speaker B is then interrupted by Speaker A. B's **I** is therefore at a turn transition but is not acknowledged. The discourse entity *our area* is then immediately deleted again from the S-List when the subsequent **I** shows that it is not part of the common ground. This means that it is not available as an antecedent for subsequent pronouns. The algorithm correctly predicts that the pronoun in A.85 does not co-specify with *our area.*

3.4 A Note on Incremental Processing

A positive feature of our model (and those such as Traum's) is that, unlike Clark & Schaefer, it allows the level of dialogue acts to be labelled incrementally. Clark & Schaefer's *Presentations* and *Acceptances* appear not only at the level of dialogue acts but at embedded levels as well, meaning that these labels can only be fully applied to the discourse as a whole.

	I	A.79	But we actually had some street people picked up last week in Dallas for picking up tin cans.
SU			
	A	B.80	My gracious.
SU	I	A.81	For picking up tin cans.
SU	I		They were going to turn them in,
	I		they were going to cash them in.
SU			
	A	B.82	Uh-huh.
	I		And they picked them up, what for?
SU			
	A/I	A.83	Disturbing the trash, or something like that.
SU			
	A	B.84	My gosh. Oh, ho , ho, ho. Oh, dear.
	~~I~~		~~Well, in our area right now~~
	I	A.85	It just blew my mind.
SU			
	A	B.86	Yes.

Figure 1: Synchronising Units and Dialogue Acts

In our model, labels at the dialogue act level (**I** and **A**) are assigned locally and incrementally, a feature which is compatible with a processing model. At the level of Synchronising Units, labels are also assigned incrementally but retrospective changes can be made. As shown in the examples above, if the content of a particular utterance indicates that the preceding utterance has been ignored, the S-List of the preceding one is deleted and the utterance not included in an **SU**.

The difference between the two levels is due to the fact that the first level represents features of the utterances themselves, whilst the second is an attempt to represent the presuppositions of both speakers. It is unlikely that the presuppositions of all participants are ever identical, so a representation of common ground can only be an approximation. Furthermore, common ground update is generally a feature of more than one utterance, meaning that immediate representation as soon as an utterance is encountered is not feasible.

4 The Algorithm

The method for resolving anaphors in spoken dialogue is based on the incremental algorithm described in Strube (1998). In our method, discourse entities are also added to the S-list (*salience list*) immediately after they are encountered. The order of the list is based on the information status of the discourse entities, basically *hearer-old* discourse entities are preferred over *hearer-new* ones (see Prince (1981) for these terms). At the end of each **SU** all discourse entities which are not realised in this **SU** are removed from the S-list. This means that the size and classification of the dialogue acts determine the set of potential antecedents of an anaphor. A major extension to the algorithm is the method for the classification of the different types of pronouns and demonstratives presented in Section 2.

The algorithm consists of two branches, one for pronouns and the other for demonstratives. Both of them call the functions *resolveInd* and *resolveDD*, which resolve individual and discourse-deictic anaphora, respectively. *resolveInd* consists only of a search through the S-list for a matching antecedent (with respect to gender and number) as described by (Strube, 1998).

resolveDD, on the other hand, consists of a search in a list containing abstract objects – the A-list, described in detail in (Eckert & Strube, 1999). Unlike the S-list, the A-list only contains objects previously referred to anaphorically. It does not contain the abstract objects referred to by each sentence and VP. This is in line with the assumption described in Section 2 that discourse-deictic anaphors themselves add a new referent to the discourse model (*referent coercion*). The A-list is necessary for multiple references to the same abstract object, as shown in example (7).

(7)	**I**	B.66	... and we make it so easy for them **[to stay there with welfare that they can get by just signing some papers.]**$_i$
			...
	I	A.75	granted, they can do **that**$_i$ very easily.
	I		**It**$_i$'s easy to do,
	I		but look where **it**$_i$ puts them. (sw2403)

In this example, we want to avoid the pronouns being resolved as referring to the preceding **I**, but rather the algorithm should co-index them with the demonstrative in A.75. The demonstrative adds the event concept entity associated with the preceding VP to the discourse model. In the algorithm, the entity is added to the A-List. The subsequent discourse-deictic pronouns look in the A-List for referents. Like the S-list, the A-list is cleaned up at the end of each **SU** – referents which were not referred to again are removed.

4.1 Context Ranking – Dialogue Acts and the Right Frontier Rule

If the A-list is empty (which is usually the case), the algorithm looks through the linguistic context for an appropriate antecedent. The order in which the possibilities are tried out is determined by the *Context Ranking* (examples are given below):

Context Ranking:

(i) A-list.

(ii) Within same **I**: Clause to the left of the clause containing the anaphor.

(iii) Within previous **I**: Rightmost main clause (and subordinated clauses to its right).

(iv) Within previous **I**'s: Rightmost complete sentence (if previous **I** is incomplete sentence).

If the A-list is empty, the algorithm looks first within the **I** containing the anaphor for the first clause to the left of the anaphor. This is successful in cases such as example (8):

(8) **I** B.104 I hope that, uh, **[they will start picking up on some of these things and, and getting involved]**$_i$, because **that**$_i$'s the only way that we're going to get out of it. (sw2403)

If there is no clause to the left, as in example (9), it looks to the previous **I** and takes the righmost main clause. Preceding main and subordinated clauses are ignored.

(9) **I** A.50 because if you tell everybody everything, **[everybody in the world would know because they'd put it on TV]**$_i$
A B.51 Right.
I A.52 and **that**$_i$ wouldn't do us any good. (sw3241)

In some cases, there is no complete main clause in the preceding **I** alone. The algorithm then looks to all preceding **I**'s until a completed main clause is found. In example (10) (an extract from Figure 1 in the previous section), Speaker A's utterance in A.83 is elliptical but the preceding question in B.82 can be used to form a syntactically complete clause.

(10) **I** B.82 And **[they picked them up, what for?**
A/I A.83 **Disturbing the trash or something like that.]**$_i$
A B.84 My gosh, Oh, ho, ho, ho. Oh dear.
Well in our area right now,
I A.85 **It**$_i$ just blew my mind. (sw2403)

Because the *Context Ranking* is expressed in terms of dialogue acts, Webber's *Right-Frontier Rule* (see Section 2) is not violated. Although the text referring to the antecedent is often not *literally* adjacent to the anaphor, it is the adjacent **I** and intervening **A**'s (B.51 in (9) and B.84 in (10)) are ignored. Unacknowledged **I**'s, i.e. those not belonging to an **SU** (B.84 in (10)) are also invisible for discourse-deictic reference.

4.2 Anaphor Classification and Resolution

A further point noted in Section 2 is that the predicative context of discourse-deictic anaphors determines what type of abstract object they refer to, i.e. whether they refer to states, events, event concepts, propositions or facts. Our algorithm at present does not have access to the semantic information necessary to make these distinctions and we assume that the predicate of the anaphor creates a referent of the correct type. However, following the notion that the predicative context gives information about the anaphor, we use it to distinguish between individual and abstract anaphors. We define that an anaphor is *I-incompatible* (cannot refer to an individual object) or *A-incompatible* (cannot refer to an abstract object) if it occurs in one of the corresponding contexts described in Table 2.

An anaphor in the object position of the verb *assume*, for example, is unlikely to have a concrete NP antecedent. This context is therefore listed as being *I-incompatible* in the table. Conversely, the object position of the verb *eat* is unlikely to have an abstract entity such as an event or a proposition as its referent, and the context is listed as *A-incompatible*.

I-Incompatible (*I) Anaphors in the x-position *cannot* refer to individual, concrete entities.	**A-Incompatible (*A)** Anaphors in the x-position *cannot* refer to abstract entities.
• Equating constructions where a pronominal referent is equated with an abstract object, e.g. *x is making it easy, x is a suggestion.* • Copula constructions whose adjectives can only be applied to abstract entities, e.g. *x is true, x is correct, x is right.* • Arguments of propositional attitude verbs which *only* take S'-complements, e.g. *assume x.* • Object of *do* (*do x*). • Predicate or anaphoric referent is a "reason", e.g. *x is because I like her, x is why he's late.*	• Equating constructions where a pronominal referent is equated with a concrete individual referent, e.g. *x is a car, x is a nice place to visit.* • Copula constructions whose adjectives can only be applied to concrete entities, e.g. *x is expensive, x is tasty, x is loud.* • Arguments of verbs describing physical contact/stimulation, which cannot be used metaphorically, e.g. *break x, smash x, eat x, drink x, smell x* but NOT **see x.*

Table 2: I-Incompatibility and A-Incompatibility

It is clear that there are problems associated with such lists. One point is that the predicates are in most cases *preferentially* associated with either abstract or individual referents rather than *categorically* (see Section 7 for a discussion of this point). This means that although a predicate may be listed as I-incompatible, it may be that an individual referent is still acceptable in some instances, and vice versa. While the lists do not reflect language use precisely, they do however greatly enhance the performance of the algorithm, as they help avoid a large number of errors.

The majority of predicates are not contained in the lists. Most predicative contexts, e.g. *know x* or *x is good*, allow both concrete and abstract referents in their argument positions. If an anaphor occurs in such a context, that is, it is neither *I-* nor *A-incompatible*, the classification depends on the success of the resolution algorithm.

The anaphora resolution algorithm is shown in Tables 3 and 4. If a pronoun (3rd person singular neuter) is encountered (Table 3), the functions *resolveDD* or *resolveInd* are evaluated, depending on whether the pronoun is *I-incompatible* (case 1) or *A-incompatible* (case 2). In the case of success the pronouns are classified as *DDPro* (discourse deictic) or *IPro* (individual), respectively. In the case of failure, the pronouns are classified as *VagPro* (vague). If the pronoun is neither *I-* nor *A-incompatible* (i.e., the predicative context of the pronoun is ambiguous in this respect), the classification is only dependent on the success of the resolution, i.e. on the availability of referents in the S/A-lists. The function *resolveInd* is evaluated first (case 3) because we observed a preference for pronouns to have individual antecedents. If successful, the pronoun is classified as *IPro*, if unsuccessful, the function *resolveDD* attempts to resolve the pronoun (case 4). If this, in turn, is successful, the pronoun is classified as *DDPro*, if it is unsuccessful it is classified as *VagPro*, indicating that the pronoun cannot be resolved using the linguistic context. The procedure is similar in the case of demonstratives (Table 4). The only difference is that (case 3) and (case 4) are reversed to capture the preference for demonstratives to be discourse-deictic (see Section 2).

3rd person masculine or feminine pronouns are resolved directly by a look-up in the S-list as

```
1. case PRO is I-incompatible
      if resolveDD(PRO)
         then classify as DDPro
         else classify as VagPro
2. case PRO is A-incompatible
      if resolveInd(PRO)
         then classify as IPro
         else classify as VagPro
3. case PRO is ambiguous
      if resolveInd(PRO)
         then classify as IPro
4.       else if resolveDD(PRO)
            then classify as DDPro
            else classify as VagPro
```

Table 3: Pronoun Resolution Algorithm

```
1. case DEM is I-incompatible
      if resolveDD(DEM)
         then classify as DDDem
         else classify as VagDem
2. case DEM is A-incompatible
      if resolveInd(DEM)
         then classify as IDem
         else classify as VagDem
3. case DEM is ambiguous
      if resolveDD(DEM)
         then classify as DDDem
4.       else if resolveInd(DEM)
            then classify as IDem
            else classify as VagDem
```

Table 4: Demonstrative Resolution Algorithm

these cannot be discourse-deictic. 3rd person plural pronouns which can be resolved this way are classified as *IPro*, if they cannot be resolved, they are marked as *IEPro* (inferrable-evoked).

4.3 An Example

The following extract from the corpus (Table 5) is used to exemplify the algorithm. The left-most column lists the **SU**'s (*28-* indicates the beginning, *-28* the end of the first **SU** in the example), the second column gives the dialogue act labels and the third the speakers and turns. For ease of representation, the S- and A-lists are only given below each **SU** in the state they are at that point in the discourse, and not everytime they are updated.

28-	I	B.18	And [**she**$_1$ ended up going to the [University of Oklahoma]$_2$]$_3$.
-28	A	A.19	Uh-huh.
			S: [DE$_1$: she, DE$_2$: Univ. of Oklahoma]
29-29	I	B.20	I can say **that**$_3$ because **it**$_2$ was a big well known school,
			S: [DE$_2$: it]
			A: [DE$_3$: that]
30-30	I		**it**$_2$ had a well known education$_4$ –
			S: [DE$_2$: it, DE$_4$: education]

Table 5: Example Analysis

At the end of **SU** 28, the S-list contains the referents of the NPs *she* and *University of Oklahoma*. The demonstrative *that* in turn B.20 is in the object position of the verb *say* and therefore classified as *I-incompatible*. The *Context Ranking* must then determine its referent. There has been no previous discourse-deictic reference so the A-list is empty (or non-existent). There is no clause in the same **I** as the anaphor so it looks to the preceding **I** and gets the referent of the main clause *she ended up going to the University of Oklahoma*. This referent is added to the A-list as *Discourse Entity*$_3$.

The first pronoun *it* in B.20 is in an A-incompatible position as the copula construction equates it with a concrete referent (*a big well-known school*). The algorithm searches through the previous S-list for the highest-ranked (only) referent – DE_2.

In **SU** 30 there is another pronoun which again is in an A-incompatible context and the S-list must be looked at for an antecedent (DE_2). Through repeated mention this referent is thus kept in the S-list for the entire length of the extract. At the end of **SU** 30 no reference has been made to the entity in the A-list (DE_3) so this list is once again empty.

5 Empirical Evaluation

Our data consisted of five randomly selected dialogues from the Switchboard corpus of spoken telephone conversations (LDC, 1993). Two dialogues were used to train the two annotators (SW2041, SW4877), and three further dialogues for testing (SW2403, SW3117, SW3241).

As a measure of inter-coder reliability we used the Kappa-statistic (Carletta, 1996). This statistic measures the percent agreement between annotators but adjusts it by the percent chance agreement for a particular classification task. This means that it takes into account the relative frequency of each class. A κ of more than .80 indicates high reliability of the classifications, a κ between .68 and .80 allows tentative conclusions, while a κ lower than .68 shows that the classification is not reliable.

Dialogue Acts. First, turns were segmented into dialogue act units. For the purpose of applying the κ statistic we turned the segmentation task into a classification task by using boundaries between dialogue acts as one class and non-boundaries as the other (see Passonneau & Litman (1997) for a similar practice). As shown in Table 6, percent agreement (PA) between the annotators was 98.35%, and $\kappa = 0.92$, indicating high reliability of the annotations.

	SW2403	SW3117	SW3241	Σ
Non-Bound.	3372	3332	1717	8421
Bound.	454	452	241	1147
N	1913	1892	979	4784
Z	1877	1866	962	4705
PA	0.9812	0.9863	0.9826	0.9835
PE	0.7908	0.7896	0.7841	0.7890
κ	0.9100	0.9347	0.9200	0.9217

Table 6: Dialogue Act Units

These dialogue act units were then classified into Initiations (I), Acknowledgments (A), Acknowledgment/Initiations (A/I), and no dialogue act (No). For this test we used only those dialogue act units which the annotators agreed about. The PA over labels given to the dialogue act units was 92.6%, $\kappa = 0.87$, again indicating that it is possible to annotate these classes reliably (Table 7).

	SW2403	SW3117	SW3241	Σ
I	230	211	108	549
A	98	120	68	286
A/I	38	41	16	95
No	0	8	8	16
N	183	190	100	473
Z	167	181	90	438
PA	0.9126	0.9526	0.9000	0.9260
PE	0.4774	0.4201	0.4152	0.4273
κ	0.8327	0.9183	0.8290	0.8708

Table 7: Dialogue Act Labels

Individual and Abstract Object Anaphora. For the classification of pronouns (IPro, DDPro, VagPro, IEPro) a PA of 87.5% was measured, $\kappa = 0.81$ (Table 8). For the classification of demonstratives (IDem, DDDem, VagDem) PA was 90.78%, $\kappa = 0.80$ (Table 9).

	SW2403	SW3117	SW3241	Σ
IPro	120	148	5	273
DDPro	33	5	9	47
VagPro	31	20	26	77
IEPro	24	20	86	130
N	104	97	63	264
Z	83	90	58	231
PA	0.7980	0.9278	0.9206	0.8750
PE	0.3935	0.6039	0.5151	0.3571
κ	0.6670	0.8170	0.8363	0.8055

Table 8: Classification of Pronouns

	SW2403	SW3117	SW3241	Σ
IDem	9	19	2	30
DDDem	45	34	28	107
VagDem	5	3	6	14
N	30	28	18	76
Z	27	26	16	69
PA	0.9000	0.9286	0.8888	0.9078
PE	0.5919	0.4866	0.6358	0.5430
κ	0.7550	0.8609	0.6949	0.7985

Table 9: Classification of Demonstratives

Co-Indexation of Anaphora. We used only those anaphors whose classification both annotators agreed upon. The annotators then marked the antecedents and co-indexed them with the anaphors. The results were compared and the annotators agreed upon a reconciled version of the data. Annotator accuracy was then measured against the reconciled version. Table 10 shows that accuracy ranged from 98.4% (Annotator A) to 96.1% (Annotator B) for individual anaphors and from 85.7% to 94.3% for abstract anaphors.

We then used the reconciled version of the annotation as the key for the individual and abstract anaphora resolution algorithms. For individual anaphors, Precision was 66.2% and Re-

			SW2403	SW3117	SW3241	Σ
Individual	**A**					
		Agreement	55	69	3	127
		No Agreement	2	0	0	2
	B					
		Agreement	56	65	3	124
		No Agreement	1	4	0	5
Discourse-deictic	**A**					
		Agreem.	31	15	14	60
		No Agreem.	7	2	1	10
	B					
		Agreem.	35	16	15	66
		No Agreem.	3	1	0	4

Table 10: Annotators' Agreement about Antecedents of Anaphora against Key

call 68.2% (Table 11), for discourse-deictic anaphors Precision was 63.6% and Recall 70% (Table 12). The low value for precision indicates that the classification did not perform very well. Only few of the anaphors resolved incorrectly were classified correctly. One of the most common errors was that a discourse-deictic or vague anaphor was classified as individual because an individual antecedent was available. A source of errors with respect to the resolution was that we did not allow the domain of the antecedent to excede one SU. However, exactly this restriction allowed us to resolve many of the discourse-deictic anaphors and also classify a high percentage of *VagPros* and *IEPros* correctly.

	SW2403	SW3117	SW3241	Σ
No. Resolved Correctly	35	52	1	88
No. Resolved Overall	50	77	6	133
No. Resolved in Key	57	69	3	129
Precision	0.7	0.675	0.167	0.662
Recall	0.614	0.754	0.333	0.682

Table 11: Results of the Individual Anaphora Resolution Algorithm

	SW2403	SW3117	SW3241	Σ
No. Resolved Correctly	25	11	13	49
No. Resolved Overall	38	19	20	77
No. Resolved in Key	38	17	15	70
Precision	0.658	0.579	0.65	0.636
Recall	0.658	0.647	0.867	0.7

Table 12: Results of the Discourse-deictic Anaphora Algorithm

6 Comparison to Related Work

Both Webber (1991) and Asher (1993) describe the phenomenon of abstract object anaphora and describe restrictions on the set of potential antecedents. They do not, however, concern themselves with the problem of how to classify a certain pronoun or demonstrative as individual or abstract. Also, as they do not give preferences on the set of potential candidates, their approaches are not intended as attempts to resolve abstract object anaphora.

To our knowledge, only little research has been carried out in the area of anaphora resolution in dialogues. LuperFoy (1992) does not present a corpus study, meaning that statistics about the distribution of individual and abstract object anaphora or about the success rate of her approach are not available. Byron & Stent (1998) present extensions of the centering model (Grosz et al., 1995) for spoken dialogue and identify several problems with the model. However, they also do not present data on the resolution of pronouns in dialogues and do not mention abstract object anaphora.

More recently, Zollo & Core (1999) presented their work on the extraction of grounding tags (which correspond to Nakatani & Traum's (1999) Common Ground Units) from dialogue tags. Their work is based on the same idea as ours, that Common Ground Units/ Synchronising Units can be derived from dialogue acts.

7 Conclusions and Future Work

We have presented an algorithm for resolving anaphors in spontaneous spoken dialogues. In particular, we have provided a method of structuring dialogues using dialogue acts to define the domain for potential antecedents, thus avoiding the problems that incomplete utterances, repetitions, false starts and utterances with no content words present for methods relying purely on syntactic units.

Furthermore, we have provided a classification system for the different types of pronouns and demonstratives found in spoken language. This makes it possible to state from the outset which ones are in principle resolvable and which ones do not have linguistic antecedents.

Only two of the pronoun and demonstrative types identified by us are resolvable. Individual anaphors, i.e. those with NP antecedents, have been dealt with by most existing algorithms. We have identified some important criteria which can be used to resolve the second type, i.e. those involving discourse deixis. Our algorithm uses information supplied by the anaphor's predicate as well as the form of the anaphor itself (pronoun vs. demonstrative) to distinguish discourse-deictic from individual reference. For the resolution process of discourse-deictic references, dialogue acts are again used to function as antecedents.

As mentioned in Section 4, our use of predicative information does not adequately reflect language use, as it generalises over preferences to make a binary distinction. While this allows the algorithm to distinguish many instances of individual and abstract anaphora, the overgeneralisation also results in some mistakes. The errors result primarily for two reasons. The first is that some verbs can be used metaphorically (e.g. *swallow*) so that seemingly A-incompatible contexts appear with reference to abstract objects, e.g., *I told him that [he'd been fired]$_i$ and he swallowed it$_i$*. Secondly, in our anaphor classification, individual anaphors are those co-indexed with NPs, and discourse-deictic anaphors are those co-indexed with VPs and clauses. This is a syntactic distinction. Our distinction between A-and I-incompatible contexts, on the other hand, is semantic, separating abstract from concrete referents. Whilst there is a correlation between NPs and concrete referents on the one hand and between clauses and abstract referents on the other, there are exceptions. Most notably, there are many NPs which refer to abstract entities, and which can therefore function as arguments in so-called A-incompatible verbal contexts, such as the event-referring subject position of *happen*, e.g., *The accident happened yesterday.*

To improve this situation, we are currently looking at the possibility of linking the algorithm to lexical databases such as WordNet (see Fellbaum (1998)) to provide it with semantic information. In WordNet, the NP *accident*, for example, is listed as a hyponym of *event*, thus explaining why it can act as an antecedent for an anaphor we predict to require an event referent.

It was pointed out in Section 2 that there are different types of abstract objects that discourse-deictic anaphors can refer to. Currently our algorithm does not distinguish between events, states, propositions and facts in the A-List. We assume, following Asher (1993), that the anaphor and its predicate select a referent of the correct type. It is clear, though, that not any clause can function as antecedent for a discourse-deictic anaphor. A clause describing a state, for example, cannot function as an antecedent for an event anaphor, e.g. **[Mary knows French.]$_i$ That$_i$ happens frequently.* We have noted in our corpus that some discourse-deictic anaphors are not immediately adjacent to their antecedents but that such anaphor-antecedent compatibility eliminates potential ambiguity. Providing the algorithm with this kind of information could be useful for selecting the correct antecedent. However, the distinction between events and states involves a complex interaction between lexical information, tense and aspect, making it difficult to determine simple rules useable in an automated process.

To our knowledge, pronoun resolution algorithms have so far not been applied to the domain of spoken language. Issues such as the number of dialogue acts functioning as the antecedent domain and the characteristics of the entities in the A-List are problems that must be solved empirically. We hope to have provided a solid basis for further work in this area by identifying the specific problems and pointing towards possible solutions.

Acknowledgments. We would like to thank Donna Byron and Amanda Stent for discussing the central issues in this paper and two anonymous reviewers for helpful comments. We are also grateful for feedback from the participants of Ellen Prince's Discourse Analysis Seminar and the audiences at the Amstelogue '99 workshop and at the Linguistics Research Department, Bell Labs, Lucent Technologies. Both authors are funded by post-doctoral fellowship awards from IRCS (NSF SBR 8920230).

References

Allen, James F. & Mark Core (1997). *DAMSL: Dialog act markup in several layers.* Draft of manual, March 1997.

Asher, Nicholas (1993). *Reference to Abstract Objects.* Dordrecht: Kluwer.

Belletti, Adriana & Luigi Rizzi (1988). Psych verbs and theta theory. *Natural Language and Linguistic Theory*, 6:291–352.

Byron, Donna & Amanda Stent (1998). A preliminary model of centering in dialog. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual*

Meeting of the Association for Computational Linguistics, Montréal, Québec, Canada, 10–14 August 1998, pp. 1475–1477.

Carletta, Jean (1996). Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254.

Carletta, Jean, Amy Isard, Stephen Isard, Jacqueline Kowtko, Gwyneth Doherty-Sneddon & Anne Anderson (1997). The reliability of a dialogue structure coding scheme. *Computational Linguistics*, 23(1):13–31.

Clark, Herbert H. & Edward F. Schaefer (1989). Contributing to discourse. *Cognitive Science*, 13:259–294.

Dahl, Östen & Christina Hellman (1995). What happens when we use an anaphor. In *Presentation at the XVth Scandinavian Conference of Linguistics Oslo, Norway.*

Eckert, Miriam & Michael Strube (1999). Resolving discourse deictic anaphora in dialogues. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics,* Bergen, Norway, 8–12 June 1999. To appear.

Fellbaum, Christiane (Ed.) (1998). *WordNet: An Electronic Lexical Database.* Cambridge, Mass.: MIT Press.

Grosz, Barbara J., Aravind K. Joshi & Scott Weinstein (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

Gundel, Jeanette K., Nancy Hedberg & Ron Zacharski (1993). Cognitive status and the form of referring expressions in discourse. *Language*, 69:274–307.

Jaeggli, Osvaldo (1986). Arbitrary plural pronominals. *Natural Language and Linguistic Theory*, 4:43–76.

LDC (1993). *Switchboard.* Linguistic Data Consortium. University of Pennsylvania, Philadelphia, Penn.

Lewis, David (1979). Keeping in a language game. In R. Baeuerle et al. (Ed.), *Semantics from a Different Point of View.* Berlin:Springer Verlag.

LuperFoy, Susann (1992). The representation of multimodal user interface dialogues using discourse pegs. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics,* Newark, Del., 28 June – 2 July 1992, pp. 22–31.

Nakatani, Christine H. & David Traum (1999). A two-level approach to coding dialogue for discourse structure: Activities of the 1998 DRI working group on higher-level structures. In *Proc. of the ACL '99 Workshop Towards Standards and Tools for Discourse Tagging, College Park, Md, June, 1999*, pp. 101–108.

Passonneau, Rebecca & Diane Litman (1997). Discourse segmentation by human and automated means. *Computational Linguistics*, 23(1):103–139.

Postal, Paul & Geoffrey Pullum (1988). Expletive noun phrases in subcategorized positions. *Linguistic Inquiry*, 19:635–670.

Prince, Ellen F. (1981). Towards a taxonomy of given-new information. In P. Cole (Ed.), *Radical Pragmatics*, pp. 223–255. New York, N.Y.: Academic Press.

Strube, Michael (1998). Never look back: An alternative to centering. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics,* Montréal, Québec, Canada, 10–14 August 1998, Vol. 2, pp. 1251–1257.

Traum, David R. (1994). *A Computational Theory of Grounding in Natural Language Conversation*, (Ph.D. thesis). Department of Computer Science, University of Rochester.

Walker, Marilyn A. (1998). Centering, anaphora resolution, and discourse structure. In M.A. Walker, A.K. Joshi & E.F. Prince (Eds.), *Centering Theory in Discourse*, pp. 401–435. Oxford, U.K.: Oxford University Press.

Webber, Bonnie L. (1991). Structure and ostension in the interpretation of discourse deixis. *Language and Cognitive Processes*, 6(2):107–135.

Zollo, Teresa & Mark Core (1999). Automatically extracting grounding tags from BF tags. In *Proc. of the ACL '99 Workshop Towards Standards and Tools for Discourse Tagging, College Park, Md, June, 1999*, pp. 109–114.

Miriam Eckert, Michael Strube
Institute for Research in Cognitive Science
University of Pennsylvania
3401 Walnut Street, Suite 400A
Philadelphia, PA 19104
USA
`{miriame|strube}@linc.cis.upenn.edu`

Belief Dynamics in Cooperative Dialogues*

Andreas Herzig and Dominique Longin†

Institut de Recherche en Informatique de Toulouse

Abstract

The context of this work is man-machine dialogues where each actor can interact with his environment, with the aim of communicating or acquiring information. We suppose that during the dialogue each participant can mistake a piece of information, forget it or simply change his mind.

We begin with some hypotheses about the actors, and present some approaches. We point out some shortcomings of the latter and introduce a new framework for belief change. We present our logical and non-logical theory; our basic notion is contextual topic: we suppose that we can associate a set of topics to every agent, speech act and formula. This allows to talk about the competence of an agent, the belief adoption and the belief preservation. We show how an agent's belief state can be define after a speech act and give a basic example.

1 Introduction

We focus on task-oriented dialogues. Participants in such dialogues have only one common goal, viz. to achieve the task under concern. Each of the participants has some information necessary to achieve the goal, but none of them can achieve it alone. For example suppose there is a system that delivers train tickets to users. The system cannot do that without user information about destination and transport class. The other way round, the user needs the system to get his ticket.

Each of the participants is supposed to be cooperative. This is a fundamental hypothesis. Informally, a person is cooperative w.r.t. another one if the former helps the latter to achieve his goals (cf. Grice's cooperation principles, as well as his conversation maxims [Grice, 1975]). Cooperation is a useful hypothesis in our context. For example, if the user wants a train ticket, then the system will intend to give it to him. If the system needs some piece of information to print the ticket, then the user answers the questions asked by the system.

Each participant is supposed to be sincere, i.e. his utterances faithfully mirror his mental state: if a participant says "the sky is blue" then he indeed believes that the sky is blue. Such a hypothesis means that contradictions cannot be explained in terms of lies. Such an assumption is much weaker than in other approaches, where sincerity is

*This work has been supported in part by the Centre National d'Études des Télécommunications (CNET) of France Telecom, in the framework of their Audiotel project.

†Thanks to Renata Wassermann, David Sadek and Jerôme Lang for relevant discussions. We also wish to thank an anonymous referee of Amstelogue'99 for his comments.

sometimes viewed as the criterion of adoption of the incoming information [Cohen and Levesque, 1990c, for example].

Under these hypotheses we aim at describing the evolution of the mental state of a rational agent participating in a conversation[1] . As we illustrate by an example below, in a conversation agents might change their mind, make mistakes, understand wrongly, ... Since the agents must interact with each other in order to achieve the goal of the dialogue (by our cooperation hypothesis), they are the victims of such phenomena. Consequently these phenomena must be taken into account when modelling the evolution of mental states. In the sequel we call *belief change* the process leading an agent from a mental state to a new one.

Probably, the most prominent formal analysis of belief change has been done in the AGM [Alchourrón *et al.*, 1985] and the KM [Katsuno and Mendelzon, 1992] frameworks. There, a belief change operator is used to define the new state from the previous one and a new piece of information (carried by an utterance). The distinction between these two is respectively based on different hypotheses on the nature of the incoming information: in the case of revision, the real world does not change (the agent just gets new information on a static world), while in the case of update it is supposed to correspond to an event that occurred in the real world. This distinction is reflected by different postulates and different semantics.

There are two reasons not to use such a framework. The first is that revision and update are incompatible [Herzig, 1998, Theorem 22], forcing us to (perhaps dynamically) choose between them. Such a choice supposes that the agent is able to recognize whether the incoming information corresponds to real world[2] change or not. Thus, if the speaker changes his mind then the hearer should update his beliefs, and if the speaker informs the hearer about his intentions (e.g. destination and transportation class in the train ticket scenario) then the hearer should revise his beliefs: he discovers the intentions of the speaker in the course of the dialogue, and he may have made wrong inferences due to misunderstanding. But it seems that he is not able in general to distinguish between these two cases.

The second reason is that both revision and update have several common properties that are not suitable in dialogues: the new information is always prioritary, the over-informing nature of some information is neglected, no distinction is made between different levels of belief (factual, introspective, alternating, ...).

Some other approaches consider each participant as a *rational agent* having *mental states* represented by different mental attitudes such as belief, knowledge, choice, goal, intention ... There, belief change takes place within a formal *rational balance theory* and a formal *rational interaction theory* [Cohen and Levesque, 1990a, 1990c]. These approaches are based on a generalization of the linguistic activity in a theory of action: they are the basis of so-called *BDI-architectures* (for Belief, Desire and Intention). Each utterance is represented by a (set of) speech act(s) [Austin, 1962, Searle, 1969], and belief change (entailed by these speech acts on the agent's mental state) must be understood and handeld as a consequence of these speech acts.

We start from the latter framework. Each participant is viewed as an autonomous agent interacting with his environment and entertaining some mental attitudes which in turn influence its behavior and its decisions. Utterances are represented by speech acts [Searle, 1969, Searle and Vanderveken, 1985]. From the agent's point of view, the dialogue is a sequence of speech acts $(\alpha_1, \dots, \alpha_n)$, where each α_{k+1} maps a mental state S_k to a new mental state S_{k+1}: nd

$$S_0 \xrightarrow{\alpha_1} S_1 \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_n} S_n.$$

S_0 is the agent's initial belief state (before the dialogue starts). Given S_k and α_{k+1}, our task is to construct the new mental state S_{k+1}. In our approach, building on previous work in [Fariñas del Cerro *et al.*, 1998], we implement belief change by an axiom of *belief adoption* and one of *belief preservation.* Both of them are based on our key concept of topics of information. We refine our previous work by contextualizing topics by mental attitudes of the agents.

The following dialogue is our running example to illustrate different problems and our solutions. There are only two agents, the system s and the user u. This example is relevant only if we take into account some laws about the world (e.g., a particular train ticket has only one destination, one class and one price, ...). In our approach, these laws are called *static laws* (Sect. 3.3):

s_1 : *Hello. What do you want?*
u_1 : *A first class train ticket to Amsterdam, please.*
s_2 : *150 €, please.*
u_2 : *Ouups! A second-class train ticket, please.*
s_3 : *100 €, please.*
u_3 : *Can I pay the 80 € by credit card?*
s_4 : *The price isn't 80 €. The price is 100 €. Yes, you can pay the 100 € by credit card.*
u_4 : ...

Within such dialogues we are particularly interested in the point of view of the machine, i.e. we study the evolution of the system's beliefs given an utterance of the user. In this case, the system:

- generally accepts some type of information (e.g. information about destination and class in train ticket selling scenarios – cf. u_1 below);
- derives supplementary information that was not directly contained in the utterance (e.g. the price if the user informs about his destination and class – s_2);
- may accept information contradicting its own beliefs such as in the case where the user changes his mind (e.g. switching from a first-class ticket to a second class ticket – u_2);
- generally preserves information it believed before the utterance (e.g. the system preserves the destination even when the class changes – from s_2 to s_3);
- may refuse to take over some information, in particular if the user tries to inform the system about facts the user isn't competent at (e.g. prices of train tickets – s_4).

We focus on the evolution of the mental state of the system s, from a hearer's point of view. s has two complementary tasks: (1) dealing with contradictions between his mental state and the new piece of information (or *input* for short), and (2) preserving his old beliefs that do not contradict this input.

In the next section we discuss the failure of the existing approaches to correctly handle such changes of mental states (Sect. 2). Then we introduce our modal framework (Sect. 3). In this framework we present an original approach based on topics (Sect. 4). Finally we illustrate the approach by a complete treatment of our running example (Sect. 5).

2 Existing approaches

In this section we review the logical analyses of belief change in dialogues that have been proposed in the literature.

2.1 Cohen and Levesque

Cohen and Levesque have defined in [Cohen and Levesque, 1990a, 1990c] a formal theory of rational interaction where an agent may accept new pieces of information. The hearer's belief adoption is conditioned by the speaker's sincerity. Their theory allows the agent both to change his beliefs, and to reject the input (if the speaker is believed to be unsincere).

In our view sincerity is not a relevant criterion for belief change. While such a hypothesis permits to explain contradictions (in terms of lies), it may lead to very naive agents: whatever the speaker utters, the hearer will believe it as soon as the speaker is believed to be sincere by the hearer.

Moreover and as Sadek notes [Sadek, 1991a], even lies might generate some effects (e.g. the hearer adds to his beliefs that the speaker is unsincere). Thus even if the input is rejected the mental state of the hearer evolves.

Finally their system says nothing about the way the beliefs not undermined by the act can be preserved from the preceding mental state to the new one. (This is nothing else the frame problem, well-known in Artificial Intelligence [McCarthy and Hayes, 1969].) Thus inconsistency of the newly acquired beliefs with old ones is never the case, just because the latter are simply given up by the agent. (Such a behaviour corresponds to what has been called the trivial belief change operation in the AGM and KM literature.)

2.2 Perrault

Perrault's system is based on Reiter's default logic [Reiter, 1980]. $A \Rightarrow B$ denotes a normal default. $Do_{\alpha,t}\top$ means that action α is performed at time t, $Observe_{j,t}$ means that agent i observes at time t, and $\langle \mathsf{Assert}_{i,j}\ P\rangle$ means that agent i communicates propositional content P to agent j.

$$\text{memory: } Bel_{i,t}A \rightarrow Bel_{i,t+1}Bel_{i,t}A \tag{1}$$
$$\text{persistence: } Bel_{i,t+1}Bel_{i,t}A \rightarrow Bel_{i,t+1}A \tag{2}$$
$$\text{observability: } Do_{\alpha,t}\top \wedge Observe_{j,t}Do_{\alpha,t}\top \rightarrow Bel_{j,t+1}Do_{\alpha,t}\top$$
$$\text{where } \alpha \text{ is performed by the agent } i \tag{3}$$
$$\text{belief transfer: } Bel_{i,t}Bel_{j,t}A \Rightarrow Bel_{i,t}A \tag{4}$$
$$\text{assertion rule: } Do_{\langle \mathsf{Assert}_{i,j}\ A\rangle,t}\top \Rightarrow Bel_{i,t}A \tag{5}$$

Moreover there is a default schema saying that if $A \Rightarrow B$ is a default then $Bel_{i,t}A \Rightarrow Bel_{i,t}B$ is also a default, for every agent i and timepoint t.

Here sincerity is not required in order to admit an act (as illustrated by the axiom (3)). But an agent consumes its effects only if he doesn't believe the converse of this effect yet (in terms of defaults: if the effect is consistent with his current beliefs, cf. (5)). Thus the speaker does not have the right to lie, to make mistakes or to change his mind, else the effect of his act will never be consumed (in technical terms, the default will be blocked).

This is at the origin of an even more radical behaviour: as highlighted in [Appelt and Konolige, 1989], Perrault's agents never question old beliefs and expand their mental state (in the sense of the AGM framework). Indeed, it follows from the axioms (1) and (2) that $Bel_{i,t}A \rightarrow Bel_{i,t+1}A$. Consequently if a belief stemming from memory conflicts with a belief stemming from the act then the default (5) will never been applied, and the effect will never be consumed.

Perrault is aware of that and suggests to achieve persistence by a default rule :

$$\text{Persistence (bis)} : Bel_{i,t+1}Bel_{i,t}A \Rightarrow Bel_{i,t+1}A \tag{6}$$

As he notes, in this case there are always two extensions: one where the agent preserves his (old) beliefs and then adopts the input if it is consistent with these beliefs, and another one where the agent adopts the input and then preserves those old beliefs that are consistent with the new information. But there seems to be no way of determine which choice the agent should make.

Perrault's approach has some other problems that we do not discuss here (e.g. if the speaker ignores whether A is the case, then he starts to believe it as soon as he utters that A, cf. [Appelt and Konolige, 1989]).

2.3 Other approaches

Several authors have proposed similar solutions. Some use monotonic logics (Sadek, Rao and Georgeff), and some nonmonotonic (Appelt and Konolige).

Sadek defines a theory of rationality similar to that of Cohen and Levesque, enriching it with two new mental attitudes, viz. uncertainty and need [Sadek, 1991a, 1992]. In his *belief reconstruction* [Sadek, 1994], he presents an alternative to Perrault's approach, enriching the latter's theory by an *axiom of admission*, and ordering the application of his axioms of memory, admission, consumption and preservation. His axiom of admission describes the behaviours that can be adopted by an agent, but does not specify the way the agent chooses between different possible behaviours. In particular he enables the hearer to reject an act. The latter point seems problematic to us, given that hearers do not reject an act that has been performed, but rather accept it in order to derive that it was not this one that has been performed. An implementation has been done at CNET (France Telecom), and make up the kernel of several applications [Sadek *et al.*, 1997].

In several papers, Rao and Georgeff have proposed theories and architectures for rational agents [Rao and Georgeff, 1991]. Such a theory can in principle be applied to dialogues. In [Rao and Georgeff, 1992], in a way similar to STRIPS, actions and plans are represented by their preconditions together with add- and delete-lists. The latter lists are restricted to sets of atomic formulas. In such a framework, one can *a priori* neither represent nondeterministic actions nor actions with indirect effects (obtained through integrity constraints). Even more importantly, actions can only have effects that are factual: this excludes the handling of speech acts, whose effects are typically represented by means of nested intensional operators (such as intentions to bring about mutual belief). Recently, they defined a tableau proof procedure for their logic [Rao and Georgeff, 1998].

Appelt and Konolige highlight the problems of Perrault's approach [Appelt and Konolige, 1989]. They propose to use hierarchic autoepistemic logic (HAEL) as a framework. Basically, what one gains there is that defaults can be stratified. This can be used to fine-tune default application and thus avoid unwanted extensions.

Apart from the relatively complex HAEL technology, it seems that Appelt and Konolige's belief adoption criterion encounters problems similar to Perrault's. Suppose the hearer has no opinion about p. If the speaker informs the hearer that p then under otherwise favourable circumstances the hearer adopts p. But if the speaker informs the hearer that *the hearer believes* p, (or that he believes the hearer believes p) then it is clearly at odds with our intuitions that the hearer should accept such an assertion about his mental state. The only means to avoid the latter behaviour seems to be to shift the hearer's ignorance about p to the prioritary level 0 of the HAEL hierarchy. But in this case the acceptation of the assertion p would be blocked as well.

3 The multi-modal framework

3.1 Language

Like the previously cited authors, we work in a multi-modal framework, with modal operators for belief, mutual belief, intention and action. Our language is that of first order multi-modal logic without equality and without function symbols [Chellas, 1980, Hughes and Cresswell, 1972, Popkorn, 1994]. We suppose that $\wedge$, $\neg$, $\top$ and $\forall$ are primitive, and that $\vee$, $\rightarrow$, $\bot$ and $\exists$ are defined as abbreviations in the usual way. Let AGT be the set of agents. For $i, j \in AGT$, the belief operators Bel_i and $Bel_{i,j}$ respectively stand for "the agent i believes that" and "the agents i and j mutually believe that". For each $i \in AGT$, the intention operator $Intend_i$ stands for "the agent i intends that". In our running example, we use two particular agents, s and u, which stand for the *system* and the *user*.

Speech acts [Austin, 1962, Searle, 1969] are represented by tuples of the form $\langle \mathsf{FORCE}_{i,j}\ A\rangle$ where FORCE is the illocutionary force of the act, $i, j \in AGT$, and A is the propositional content of the act. For example $\langle \mathsf{Inform}_{u,s}\ Dest(\text{Amsterdam})\rangle$ represents a declarative utterance of the user informing the system that the destination of his ticket is Amsterdam. Let ACT be the set of all speech acts.

With every speech act $\alpha \in ACT$ we associate two modal operators $Done_\alpha$ and $Feasible_\alpha$. $Done_\alpha A$ is read "speech act α has just been performed, before which A was true"; $Feasible_\alpha A$ is read "speech act α is feasible, after which A will be true".[3] In particular, $Done_\alpha \top$ and $Feasible_\alpha \top$ are respectively read "α has just been performed" and "α is feasible" (or "can be performed"). Using the $Done_\alpha$ operator, the beliefs of the system at the mental state S_k can be kept in *memory* at state S_{k+1}. (If B is the conjunction of all beliefs of the agent i at the (mental) state k, and α has just been done, then $Bel_i Done_\alpha B$ is the memory of i in the state $k+1$.) Formally, acts and formulas are defined by mutual recursion. (This enables speech acts where the propositional contents is a non-classical formula.) For example, $Bel_s Done_{\langle \mathsf{Inform}_{u,s}\ Bel_u Bel_s p\rangle} Bel_s Bel_u Bel_s \neg p$ is a formula.

To express temporal properties, we define the *Always* operator, and his dual operator *Sometimes*. $Always A$ means "A always holds" and $Sometimes A$ means "A sometimes holds". The operator *Always* allows in particular to preserve the laws in all mental states.

3.2 Axioms

Just like in [Cohen and Levesque, 1990b, Perrault, 1990, Sadek, 1991a], with each belief operator we associate the (normal) modal logic KD45 [Halpern and Moses, 1985]. Thus, for all agent $i \in ACT$ and all formulas A and B:

$$\frac{A}{Bel_i A} \tag{RN$_{\text{Bel}}$}$$

$$Bel_i A \wedge Bel_i(A \rightarrow B) \rightarrow Bel_i B \tag{K$_{\text{Bel}}$}$$

$$Bel_i A \rightarrow \neg Bel_i \neg A \tag{D$_{\text{Bel}}$}$$

$$Bel_i A \rightarrow Bel_i Bel_i A \tag{4$_{\text{Bel}}$}$$

$$\neg Bel_i A \rightarrow Bel_i \neg Bel_i A \tag{5$_{\text{Bel}}$}$$

are logical axioms: (RN$_{\text{Bel}}$) and (K$_{\text{Bel}}$) are axioms of every normal modal logic, (D$_{\text{Bel}}$) is the "axiom of rationality" (if i believes A then he doesn't believes $\neg A$), (4$_{\text{Bel}}$) is the positive introspection (if i believes A then he believes that he believes A), and (5$_{\text{Bel}}$) is the negative introspection (if i doesn't believe A then he believes that he doesn't believes A).

With each operator of mutual belief we associate the (normal) modal logic KD45 whose logical axioms are similar to these of belief operators. We suppose that common belief is related to belief by the logical axiom

$$Bel_{i,j} A \rightarrow Bel_i A \tag{7}$$

To keep things simple we suppose that the logic of each operator of intention is the normal modal logic KD. (The axioms (RN$_{\text{Intend}}$), (K$_{\text{Intend}}$) and (D$_{\text{Intend}}$) are similar to (RN$_{\text{Bel}}$), (K$_{\text{Bel}}$) and (D$_{\text{Bel}}$).)

Obviously, our notions of common belief and intention are oversimplified: first, our condition linking belief and common belief is weaker than the usual induction axiom. We argue that such an inductive principle is not necessary at least in a first approach: as Cohen and Levesque, we suppose that common belief directly comes as the indirect effect of a speech act. (This is different from Perrault's view, where mutual belief is constructed via default rules.) Second, we offer no particular principle for intentions. We did this because the existing analyses of intention vary a lot, and the systems that have been put forward in the literature are rather complex. A normal modal logic for intention is too strong: for example, (K$_{\text{Intend}}$) is not a theorem of Cohen and Levesque's logic (and neither is its converse). However, we think we are able to define intention operators in a minimal models semantics [Chellas, 1980, Chap. 7].

All $Done_\alpha$ and $Feasible_\alpha$ operators obey the principles of the (normal) modal logic K. As they are modal operators of "possible" type, the rule of necessitation and the K-axiom take the form:

$$\frac{\neg A}{\neg Done_\alpha A} \tag{RN$_{\text{Done}}$}$$

$$(\neg Done_\alpha A \wedge Done_\alpha B) \rightarrow Done_\alpha(\neg A \wedge B) \tag{K$_{\text{Done}}$}$$

$$\frac{\neg A}{\neg Feasible_\alpha A} \tag{RN$_{\text{Feasible}}$}$$

$$(\neg Feasible_\alpha A \wedge Feasible_\alpha B) \rightarrow Feasible_\alpha(\neg A \wedge B) \tag{K$_{\text{Feasible}}$}$$

We want speech acts to be deterministic in order that performing it should lead to a single mental state. It is classical to represent such a determinism by the converse of the modal axiom (D). Thus, for all $\alpha \in ACT$ and every formula A:

$$Done_\alpha A \rightarrow \neg Done_\alpha \neg A \tag{DC$_{\text{Done}}$}$$

$$Feasible_\alpha A \rightarrow \neg Feasible_\alpha \neg A \tag{DC$_{\text{Feasible}}$}$$

The following logical axioms illustrate the interaction between the $Done_\alpha$ and $Feasible_\alpha$ operators:

$$Done_\alpha \neg Feasible_\alpha A \rightarrow \neg Feasible_\alpha Done_\alpha A \quad (8)$$
$$Feasible_\alpha \neg Done_\alpha A \rightarrow \neg Done_\alpha Feasible_\alpha A \quad (9)$$

The logic of the *Always* operator is the normal modal logic KT4. ($\mathrm{K}_{\mathrm{Time}}$) and ($4_{\mathrm{Time}}$) are similar to ($\mathrm{K}_{\mathrm{Bel}}$) and ($4_{\mathrm{Bel}}$):

$$Always A \rightarrow A \quad (\mathrm{T}_{\mathrm{Time}})$$

and

$$Sometimes A \stackrel{\mathrm{def}}{=} \neg Always \neg A \quad (\mathrm{Def}_{\mathrm{Sometimes}})$$

In order to describe some interactions between the different mental attitudes [Cohen and Levesque, 1990b], we propose the following logical axioms. For all agent $i, j \in AGT$ and every formula A:

$$Intend_i A \rightarrow Intend_i Bel_i A \quad (10)$$
$$Bel_i Intend_i A \leftrightarrow Intend_i A \quad (11)$$
$$Bel_i \neg Intend_i A \leftrightarrow \neg Intend_i A \quad (12)$$
$$Intend_i Bel_j A \rightarrow Bel_i A \vee Intend_i Bel_i A \quad (13)$$
$$Bel_i Done_{\langle \mathsf{F}_{i,j}\ A \rangle} \top \leftrightarrow Done_{\langle \mathsf{F}_{i,j}\ A \rangle} \top \quad (14)$$

We give the semantics of every logical axiom in [Longin, 1999].

3.3 Non-logical laws

The laws are non-logical axioms. These axioms musn't be modified by the belief change process. They are preserved in every mental state by the *Always* operator. Let *LAWS* be the set of all laws.

Our non-logical theory contains tree kinds of laws: *static laws* (similar to integrity constraints in data bases); *laws governing speech acts* (to describe the different preconditions and effects of the speech acts); *reactive laws* (to describe some reactive behaviours generating intentions).

Static laws. Some of the static laws are believed only by the system. It's the case for the prices of tickets according to the destination and the class, for example:

$$Always Bel_s(Dest(\text{Amsterdam}) \wedge Class(\text{1st}) \rightarrow Price(150\,€)) \quad (15)$$
$$Always Bel_s(Dest(\text{Amsterdam}) \wedge Class(\text{2nd}) \rightarrow Price(100\,€)) \quad (16)$$
$$\ldots$$

Some others are knew both by the system and the user:

$$Always Bel_{i,j} \neg(Class(\text{1st}) \wedge Class(\text{2nd})) \quad (17)$$
$$Always Bel_{i,j} \neg(Dest(\text{Toulouse}) \wedge Dest(\text{Amsterdam})) \quad (18)$$
$$\ldots$$

Laws governing speech acts. Following Sadek [Sadek, 1991b], we associate with each speech act

- a precondition;
- an indirect effect (viz. the persistence of preconditions after the performance of the speech act);
- an intentional effect (in the Gricean sense [Grice, 1967]);
- a perlocutionary effect (expected effect).

PRECONDITIONS are represented by formulas of the form $AlwaysBel_k \neg Done_\alpha \neg A'$ where A' is a precondition of α, and k an agent (there is no constraint on k: it may be the speaker, or a hearer). For example, such a law for informative acts is (preconditions and effects of our speech acts follows from [Sadek, 1991a, 1991b]):

$$\begin{aligned} AlwaysBel_k \neg Done_{\langle \mathsf{Inform}_{i,j}\ A\rangle} \neg (&Bel_i A \wedge \neg Bel_i Bel_j \neg Bel_i A \wedge \\ &\neg Bel_i BelIf_j A \wedge \neg Bel_i Bel_j BelIf_j A) \end{aligned} \tag{19}$$

where $BelIf_j A$ is an abbreviation for $Bel_j A \vee Bel_j \neg A$. The precondition means: the agent i believes A, and he doesn't believes that j believes that he doesn't believes A (sincerity condition[4]), and he doesn't believes that j knows if A holds or not, and he doesn't believes that j believes that j knows if A holds or not (condition of relevance to the context[5]).

From this law and the standard principles of our logic, it follows formulas of the form $AlwaysBel_k(Done_\alpha \top \rightarrow Done_\alpha A')$ where A' is a precondition of α. For example:

$$\begin{aligned} AlwaysBel_k(Done_{\langle \mathsf{Inform}_{i,j}\ A\rangle} \top \rightarrow Done_{\langle \mathsf{Inform}_{i,j}\ A\rangle}(&Bel_i A \wedge \\ &\neg Bel_i Bel_j \neg Bel_i A \wedge \neg Bel_i BelIf_j A \wedge \\ &\neg Bel_i Bel_j BelIf_j A)) \end{aligned} \tag{20}$$

Suppose that the user informs the system he would like a first class ticket. Then:

1. $Bel_s Done_{\langle \mathsf{Inform}_{u,s}\ Class(1st)\rangle} \top$ (the act has just been performed) ;
2. $Bel_s Done_{\langle \mathsf{Inform}_{u,s}\ Class(1st)\rangle}(Bel_u Class(1\text{st}) \wedge \neg Bel_u Bel_s \neg Bel_u Class(1\text{st}) \wedge$ $\neg Bel_u BelIf_s Class(1\text{st}) \wedge \neg Bel_u Bel_s BelIf_s Class(1\text{st}))$
 (by (20) where $k = s$, 1., and principles of our logic);
3. $Bel_s Done_{\langle \mathsf{Inform}_{u,s}\ Class(1st)\rangle}(Bel_u Class(1\text{st}) \wedge \neg Bel_u Bel_s \neg Bel_u Class(1\text{st}) \wedge$ $\neg Bel_u BelIf_s Class(1\text{st}) \wedge \neg Bel_u Bel_s BelIf_s Class(1\text{st}))$
 (by (20) where $k = u$, 1., and principles of our logic).

The formulas 2. and 3. are what we call *presuppositions*: when a speech act has just been performed, an agent believes now that the preconditions of this act were true for the speaker just before the performance of this act.

THE INDIRECT EFFECT is the preservation of preconditions, and must be derived from presuppositions by formulas of the form $AlwaysBel_k(Done_\alpha A' \rightarrow A')$ where A' is the preconditions of α: this is a particular instance of our axiom schema of belief preservation (cf. Sect. 4.2).

THE INTENTIONAL EFFECT is the consumption by the hearer of the intention of the speaker (in the sens of Grice). Such a consumption is represented by formulas of the form

$AlwaysBel_k(Done_\alpha \top \rightarrow A'')$ where A'' is the intentional effect of α. For example, an instance of this schema for an informative speech act is:

$$AlwaysBel_k(Done_{(\mathsf{Inform}_{i,j}\ A)} \top \rightarrow Intend_i Bel_j Intend_i Bel_j A) \tag{21}$$

THE PERLOCUTIONARY EFFECT is not consumed (there is no schema to add it to the mental state of an agent): our agents are autonomous (the expected effect of an act doesn't happen systematically). If the new mental state, obtained by the admission of a speech act and the consumption of his indirect and intentional effects, entails the perlocutionary effect, we said that the latter has been consumed (but it is a misuse of language).

Reactive laws. The reactive laws allow us to generate some intentions:

$$AlwaysBel_i(A \wedge Bel_j \neg A \rightarrow Intend_i Bel_j A) \tag{22}$$

$$AlwaysBel_i(A \wedge Done_{(\mathsf{Inform}_{j,i}\ A)} Bel_i \neg A \rightarrow Intend_i Bel_j Bel_i A) \tag{23}$$

$$AlwaysBel_i(Done_\alpha(Done_\gamma \top \wedge Bel_i Done_\beta \top) \rightarrow Intend_i Bel_j Bel_i Done_\alpha Done_\gamma \top) \tag{24}$$

...

For example, (22) is used for the first part of the utterance s_4 of our running example: the system invalidates the price of 80€, and informs the user that the price is 100€. Formally (we don't give the logical axioms we use, s and u are respectively the agents i and j in the law (22)):

1. $Bel_s Price(80€)$ (hypothesis)
2. $Bel_s Bel_u Price(100€)$ (hypothesis)
3. $Bel_{s,u} \neg(Price(80€) \wedge Price(100€))$ (static law)
4. $Bel_s \neg Price(100€)$ (by 1. and 3.)
5. $Intend_s Bel_u \neg Price(100€)$ (by (22), 4. and 2.)
6. $Bel_s Bel_u \neg Price(80€)$ (by 2. and 3.)
7. $Intend_s Bel_u Price(80€)$ (by (22), 1. and 6.)

The intentions in 5. and 7. are respectively associated with a deny speech act (the price isn't 80€) and an informational speech act (the price is 100€).

3.4 The problem of belief change

In our approach, unlike to Sadek we always accept[6] speech acts, but we proceed in two steps: the agent accepts the indirect and intentional effects, but only adopts the speaker's beliefs if he believes the speaker to be competent about these beliefs. Thus, the speaker's competence is our criterion to determine which part of the input must be accepted by the hearer and which part must be rejected. For example, s accepts the input about the new class (after u_2) but rejects the input about the price (after u_3), the reason being that he considers u to be competent in classes but not in ticket prices. Which beliefs of the hearer can be preserved after the performance of a speech act? Our key concept here is that of the *influence of a speech act on beliefs.* If there exists a relation of influence between the speech act and a belief, this belief cannot be preserved in the new mental state. (In our example, the old transport class cannot be preserved through u_2, because the act of informing about classes influences the hearer's beliefs about classes.)

All this presupposes that we are able to determine the competence of an agent and the influence relationship between speech acts and beliefs. The foundation for both notions will be provided here by the concept of a *topic*: we start from the idea that with every agent, speech act, and formula, some set of topics can be associated. Thus, an agent is competent about a formula A if and only if the set of topics associated with A is a subset of the set of topics associated with this agent – the set of topics the agent is competent in. And a formula A is preserved after the performance of a speech act α if there is no topic that occurs both in the set of topics associated to A and in the set of topics associated to the speech act. The next section gives the formal apparatus.

4 Topic-based belief change

To describe the relation between a formula and a speech act, and the relation between a formula and an agent, we use the concept of topic. This notion is very important from a linguistic and a logical point of view.

For example, in [Büring, 1995], a semantical value related to the topics is associeted with each English sentence. Van Kuppevelt has developed a notion of topic based on questions, and has applied it to phenomena of intonation [van Kuppevelt, 1991, 1995]. In [Ginzburg, 1995], some sets of topics play a decisive role in the coherence of dialogues.

Epstein (1990) defines the *relatedness relation* $\mathcal{R}$ as a primitive relation between propositions because "the subject matter of a proposition isn't so much a property of it as a relationship it has to other propositions" [Epstein, 1990, page 62]. Thus, he does not represent topics explicitly. Then he defines the *subject matter of a proposition* A as $s(A) = \{\{A, B\} : \mathcal{R}(A, B)\}$. More precisely, s is called the *subject matter set-assignment associated with* $\mathcal{R}$ [Epstein, 1990, page 68]. Epstein shows that we can also define s as primitive, and that we can then define two propositions as being related if they have some subject matter in common. Our $\mathfrak{subject}$ function can be seen as an extension of this function to a multi-modal language.

Other studies of the notion of topic exist in the literature, in particular those of Lewis [Lewis, 1972] and Goodman [Goodman, 1961]. Both are quite different from Epstein's. Goodman's notion of "absolute aboutness" is defined purely extensionally. Hence for him logically equivalent formulas are about the same topics, while this is not the case for us. Moreover, as he focusses on the "informative aspect" of propositions, the subject of a tautology is the empty set.

4.1 Topic structures

4.1.1 Themes and topics

A *theme* is what something is about. For example, an information on the destination is about the destination but not about the transport class.

Let $\mathcal{T} \neq \emptyset$ be a set that we call the set of themes. In our running example, we suppose that destinations, classes, and prices are among the themes.

Definition 1 *Let $i \in AGT$. Then ma_i is called an* atomic context. *A* context *is a possibly empty sequence of atomic contexts. The empty context is noted λ. $\mathcal{C}$ is the set of all the contexts.*

ma_i stands for "the mental attitude of agent i". In this paper we shall suppose that the length of each context is at most 2.

Definition 2 *A* topic of information *(or* contextual thematic structure*) is a theme together with a context denoted by $c{:}t$ where $t \in \mathcal{T}$, $c \in \mathcal{C}$, and c is of length at most 2. Given a set of themes $\mathcal{T}$ we note $\mathbb{T}$ the associated set of topics.*

$\lambda{:}c = c{:}\lambda = c$. By convention, $\lambda{:}t$ is abbreviated by t. Hence $\mathcal{T} \subseteq \mathbb{T}$. For example, $ma_u{:}price$ is a topic consisting in the user's mental attitude at prices, and $ma_s{:}ma_u{:}price$ is a topic consisting in the system's mental attitude at the user's mental attitude at prices.

4.1.2 The subject of a formula

Definition 3 *We call* subject of a formula *a set of topics the formula is about. This notion is formalized by a function* $\mathfrak{subject}$ *mapping a formula to a set of topics from* $\mathbb{T}$.

The spirit of our $\mathfrak{subject}$ function is that of Epstein. We give the following axioms.

Axiom 1 $\mathfrak{subject}(p) \subseteq \mathcal{T}$ *and* $\mathfrak{subject}(p) \neq \emptyset$ *where p is atomic.*

Axiom 2 $\mathfrak{subject}(\top) = \emptyset$.

(viz. The truth is about nothing.)[7]

Axiom 3 $\mathfrak{subject}(\neg A) = \mathfrak{subject}(A)$.

Axiom 4 $\mathfrak{subject}(A \wedge B) = \mathfrak{subject}(A) \cup \mathfrak{subject}(B)$.

Axiom 5

$$\mathfrak{subject}(Bel_i A) = \{ma_i{:}c{:}t \mid c \neq ma_i \text{ and } (c{:}c'{:}t \in \mathfrak{subject}(A) \text{ or } ma_i{:}c{:}t \in \mathfrak{subject}(A))\}.$$

Note that c might be the empty context here. Thus, in our running example:

$$\begin{aligned}
&\mathfrak{subject}(Class(\text{1st})) = \{classe\} \\
&\mathfrak{subject}(Dest(\text{Nice})) = \{destination\} \\
&\mathfrak{subject}(Bel_s Bel_u Price(80\,€) \wedge Bel_s Price(100\,€)) = \{ma_s{:}ma_u{:}prix\} \cup \{ma_s{:}prix\}.
\end{aligned}$$

Theorem 1 *If A is a formula that contains no modal operator, then:*

$$\begin{aligned}
\mathfrak{subject}(Bel_i A) &= \mathfrak{subject}(Bel_i ... Bel_i A) \\
&= \{ma_i{:}t \mid t \in \mathfrak{subject}(A)\}.
\end{aligned} \tag{25}$$

$$\begin{aligned}
\mathfrak{subject}(Bel_i Bel_j A) &= \mathfrak{subject}(Bel_i ... Bel_i Bel_j ... Bel_j Bel_k ... A) \\
&= \{ma_i{:}ma_j{:}t \mid i \neq j, t \in \mathfrak{subject}(A)\}.
\end{aligned} \tag{26}$$

Axiom 6 $\mathfrak{subject}(Intend_i A) = \mathfrak{subject}(Bel_i A)$.

Axiom 7 $\mathfrak{subject}(Done_\alpha A) = \mathfrak{subject}(A) \cup \mathfrak{subject}(A')$ *where A' is the propositional content of α.*

Axiom 8 $\mathfrak{subject}(\forall x A) = \mathfrak{subject}(A)$.

Axiom 9 $\mathfrak{subject}(A[t/x]) \subseteq \mathfrak{subject}(A)$, *where $A[t/x]$ is the formula resulting from the substitution of the variable x by the term t.*

Our $\mathfrak{subject}$ function is not extensional: logically equivalent formulas may have different topics. (An intuition that might be helpful is to think of the subject of A as the set of predicates names occurring in A).

4.1.3 The competence of an agent

Now we associate topics to agents.

Definition 4 *We call* competence of an agent *the set of topics in which this agent is competent. This notion is formalized by a function* $\mathfrak{competence}$ *mapping an agent to a set of topics from* $\mathbb{T}$.

We assume every agent is competent in his mental states.

Axiom 10 $\mathfrak{competence}(i) \supseteq \{ma_i{:}t \mid t \in \mathcal{T}\}$.

An agent may be competent about some facts. For example, $\mathfrak{competence}(u)$ contains destinations and classes, but not prices.[8]

Competence will allow us to formulate in the next section our belief adoption axiom which basically says: "an agent j adopts the belief of an other agent i about a formula A if j considers that i is competent in the subject of A. [9] "

4.1.4 The scope of an act

In the formalization of speech acts the illocutionary force determines a set of formula schemes (the preconditions and the effects of the act) instantiated by the propositional content. The scope of a speech act is the set of topics associated with this act, and must depend on its illocutionary force and its propositional content.

Roughly speaking, the themes of a speech act are determined by its propositional content, and the context by its illocutionary force. Thus, contexts tell us which mental attitudes might change.

Definition 5 *We call* scope of a speech act *a set of topics associated to this speech act. This notion is formalized by a function* $\mathfrak{scope}$ *mapping a speech act to a set of topics from* $\mathbb{T}$.

The scope of a speech act determines what mental attitudes of an agent are questioned by this act.

We propose some axioms in order to compute the scope of a speech act. The performance of a speech act always influences some mental attitudes of the hearer. In particular:

Axiom 11 $\mathfrak{scope}(\langle \mathsf{FORCE}_{i,j}\ A\rangle) \supseteq \{ma_j{:}ma_i{:}t \mid t \in \mathfrak{subject}(A)\}$, *for every illocutionary force* FORCE.

In the case of request, we postulate that these mental attitudes are the only one questioned.

For example, consider the speech act where the user informs the system about the ticket price. This speech act influences the system's belief about the user's belief about prices.

4.1.5 Topic structures

We have thus defined three functions mapping the different types of expressions in our language to topics.

Definition 6 *Given a set of themes, a* topic structure *consists of the associated set of topics together with the* $\mathfrak{subject}$, $\mathfrak{scope}$, *and* $\mathfrak{competence}$ *functions.*

Is there an interaction between these functions? We propose the following axiom for acts of the informative type.

Axiom 12 *If A is factual, $\alpha = \langle \mathsf{Inform}_{i,j}\, A \rangle$, and t is a theme such that $t \in \mathfrak{subject}(A) \cap \mathfrak{competence}(i)$ then $t \in \mathfrak{scope}(\alpha)$ and $ma_j{:}t \in \mathfrak{scope}(\alpha)$.*

For example, if the user informs the system about his destination, as the user is competent at destinations, then this speech act influences the system's factual beliefs about the destination, and also about prices, because a change in the destination possibly entails a change in the price: $\mathfrak{scope}(\langle \mathsf{Inform}_{u,s}\, Dest(\text{Amsterdam}) \rangle)$ contains the topics *destination*, *price*, ma_s*:destination* and ma_s*:price*.

A given topic structure will allow us to reconstruct beliefs by means of two principles: competence and preservation. In the next section we shall present these principles.

4.2 Axioms for belief change

Our axioms for belief change are based on a given topic structure. The first one is the axiom schema of belief preservation:

Axiom Schema of Belief Preservation.

$$Done_\alpha A \to A \text{ if } \begin{cases} \mathfrak{scope}(\alpha) \cap \mathfrak{subject}(A) = \emptyset \text{ and} \\ A \text{ contains no } Done_\beta \text{ operator.} \end{cases}$$

The restriction to formulas without $Done_\beta$ operator is necessary because our reading of $Done_\beta$ is that β has just been performed (and not at some arbitrary time point in the past).

The second axiom schema is about the adoption of beliefs.

Axiom Schema of Belief Adoption.

$$Bel_i A \to A \text{ if } \mathfrak{subject}(A) \subseteq \mathfrak{competence}(i)$$

The schema expresses that: if the agent i believes that A, and if he's comptent about A, then A is true.

For example the formula $Bel_s Bel_u Dest(\text{Amsterdam}) \to Bel_s Dest(\text{Amsterdam})$ can be proved from the instance $Bel_u Dest(\text{Amsterdam}) \to Dest(\text{Amsterdam})$ of the belief adoption axiom, because $\mathfrak{subject}(Dest(\text{Amsterdam})) \subseteq \mathfrak{competence}(u)$. (By using the standard modal necessitation and K-principles for Bel_s). On the contrary, $Bel_u Price(80€) \rightarrow Price(80€)$ is not an instance of our axiom schema, because $\mathfrak{subject}(Price(80€)) \not\subseteq \mathfrak{competence}(u)$.

Remark 1 *In our preceding approach [Fariñas del Cerro* et al., *1998] we had used non-contextualized topics to formulate axioms for belief change. This turned out to be too weak. Suppose the system believes p, and believes that the user believes p: $Bel_s p \wedge Bel_s Bel_u p$. Now suppose the user informs the system that he does not know whether p. Then the belief $Bel_s Bel_u p$ should go away, while $Bel_s p$ can be expected to be preserved. Hence the scope of this speech act should contain the system's attitudes about the users's attitudes about p, but not the system's attitudes about p. We were not able to distinguish that before.*

5 Example

We illustrate our belief change process by our running example. To each uterrance number, we associate a speech act number (e.g. α_{u_i} points at the u_i utterance). We describe the different mental states S_{s_i} of the system after the different speech acts of the user. S_{s_0} is the initial state (supposed empty). Thus, the different mental states that we want to describe are the following:

$$S_{s_0} \xrightarrow{\alpha_{s_1}} S_{s_1} \xrightarrow{\alpha_{s_2}} \dots \xrightarrow{\alpha_{s_n}} S_{s_n}$$

The different speech acts performed are respectively:
$\alpha_{u_1} = \langle \mathsf{Inform}_{u,s}\ Class(\text{1st}) \wedge Dest(\text{Nice})\rangle$
$\alpha_{u_2} = \langle \mathsf{Inform}_{u,s}\ Class(\text{2nd})\rangle$
$\alpha_{u_3} = \langle \mathsf{Inform}_{u,s}\ Price(80€)\rangle; \langle \mathsf{ReqInformIf}_{u,s}\ Payment(\text{carte de crédit})\rangle$
The different scopes corresponding to these speech acts are:

- $\mathfrak{scope}(\alpha_{u_1}) \supseteq \{ma_s{:}ma_u{:}t, ma_s{:}t\}$ where t is a theme of the propositional content of α_{u_1} ($t \in \{class, destination, \dots\}$);
- $\mathfrak{scope}(\alpha_{u_2}) \supseteq \{ma_s{:}ma_u{:}t, ma_s{:}t\}$ where t is a theme of the propositional content of α_{u_2} ($t \in \{class, \dots\}$);
- $\mathfrak{scope}(\alpha_{u_3}) \supseteq \{ma_s{:}ma_u{:}t\}$ where t is a theme of the propositional content of each speech act of α_{u_3} ($t \in \{price, payment, \dots\}$);
- $\mathfrak{competence}(u) \supseteq \{ma_u{:}t \mid t \in \mathcal{T}\} \cup \{destination, class\}$;
- $\mathfrak{competence}(s) \supseteq \{ma_s{:}t \mid t \in \mathcal{T}\} \cup \{price, payment\}$;.

We use the following abbreviations:

- $S_{s_i}^{-Done}$ is the conjunction of formulas of the state S_{s_i} which contain no *Done*-operator;
- $C1$ and $C2$ are respectively $Class(\text{1st})$ and $Class(\text{2nd})$;
- D is $Dest(\text{Nice})$;
- $P1$, $P2$ and $P3$ are respectively $Price(150€)$, $Price(100€)$ and $Price(80€)$ (bad price);

- CC is *Payment*(credit card).

We do not give every formula produced by our logic. The preconditions of the speech acts are simplifyed. Thus, the different mental states are the following:

$S_{s_0} = \emptyset$

S_{s_1} contains the following formulas:

1. performance of the act: $Bel_s Done_{\alpha_{u_1}} \top$
2. presuppositions: $Bel_s Done_{\alpha_{u_1}} (Bel_u (C1 \wedge D) \wedge \neg Bel_u BelIf_s (C1 \wedge D))$
3. indirect effect: $Bel_s Bel_u C1 \wedge Bel_s Bel_u D \wedge Bel_s \neg Bel_u BelIf_s (C1 \wedge D)$
4. intensional effect: $Bel_s Intend_u Bel_s Intend_u Bel_s C1 \wedge$ $Bel_s Intend_u Bel_s Intend_u Bel_s D$
5. reduction of intention: $Bel_s Intend_u Bel_s C1 \wedge Bel_s Intend_u Bel_s D$
6. adoption: $Bel_s C1 \wedge Bel_s D$
7. static laws: $Bel_s P1$

S_{s_2} contains the following formulas:

1. performance of the act: $Bel_s Done_{\alpha_{u_2}} \top$
2. memory: $Bel_s Done_{\alpha_{u_2}} S_{s_1}^{-Done}$
3. presuppositions: $Bel_s Done_{\alpha_{u_2}} (Bel_u C2 \wedge \neg Bel_u BelIf_s C2)$
4. indirect effect: $Bel_s Bel_u C2 \wedge Bel_s \neg Bel_u BelIf_s C2$
5. intensional effect: $Bel_s Intend_u Bel_s Intend_u Bel_s C2$
6. reduction of intention: $Bel_s Intend_u Bel_s C2$
7. preservation: $Bel_s Bel_u D \wedge Bel_s Intend_u Bel_s Intend_u Bel_s D \wedge$ $Bel_s Intend_u Bel_s D \wedge Bel_s D$
8. adoption: $Bel_s C2$
9. static laws: $Bel_s P2$

S_{s_3} contains the following formulas:

1. performance of the act: $Bel_s Done_{\alpha_{u_3}} \top$
2. memory: $Bel_s Done_{\alpha_{u_3}} S_{s_2}^{-Done}$
3. presuppositions: $Bel_s Done_{\alpha_{u_3}} (Bel_u P3 \wedge \neg Bel_u BelIf_s P3 \wedge \neg BelIf_u CC \wedge$ $Bel_u \neg Intend_s Done_{\langle \mathsf{InformIf}_{s,u}\, CC \rangle} \top)$
4. indirect effect: $Bel_s Bel_u P3 \wedge Bel_s \neg Bel_u BelIf_s P3 \wedge Bel_s \neg BelIf_u CC \wedge$ $Bel_s Bel_u \neg Intend_s Done_{\langle \mathsf{InformIf}_{s,u}\, CC \rangle} \top)$

5. intensional effect: $Bel_s Intend_u Bel_s Intend_u Bel_s P3 \wedge$
$Bel_s Intend_u Bel_s Intend_u Done_{\langle \mathsf{InformIf}_{s,u}\, CC \rangle} \top$

6. reduction of intention: $Bel_s Intend_u Bel_s P3 \wedge$
$Bel_s Intend_u Done_{\langle \mathsf{InformIf}_{s,u}\, CC \rangle} \top$

7. preservation: $Bel_s Bel_u C2 \wedge Bel_s \neg Bel_u BelIf_s C2$
$\wedge Bel_s Intend_u Bel_s Intend_u Bel_s C2 \wedge Bel_s Intend_u Bel_s C2 \wedge Bel_s Bel_u D \wedge$
$Bel_s Intend_u Bel_s Intend_u Bel_s D \wedge Bel_s Intend_u Bel_s D \wedge Bel_s D \wedge Bel_s C2$

8. static laws: $Bel_s P2$

9. reactive laws: $Intend_s Bel_u \neg P3 \wedge Intend_s Bel_u P2$

The aim of our approach is always to generating a consistent mental state. Formally, we can describe how the new mental state S_{k+1} is constructed from the state S_k:

$$S_{k+1} = \{Bel_i Done_{\alpha_{k+1}} A : A \in S_k\} \cup LAWS.$$

Hence, $S_{k+1} \to C$ ssi $LAWS \vdash (Bel_i Done_{\alpha_{k+1}} Bel_i Done_{\alpha_k} \ldots Bel_i Done_{\alpha_1} S_0) \to C$.

In the below example, our propositional contents are relatively simple: however, there exists some laws which permit to treat more complex propositional contents [Sadek, 1991a, Longin, 1999].

6 Conclusion

We have sketched a theory of change in the context of dialogues. It is based on the notion of topic of information, which is exploited through topic-based axioms of belief adoption and preservation.

Perrault and Appelt & Konolige have argued that defaults are crucial elements in a theory of speech acts because in a way they permit to transform absence of knowledge into knowledge. In a sense, what we do is to transferring that task to the metalinguistic relations of competence and scope. This permits to keep a monotonic framework.

We have supposed that the set of topics associated with a formula is determined by those of the atomic formulas occurring in it. This is certainly a debatable choice. It was mainly motivated by representational economy. Notwithstanding, the way we use the $\mathfrak{subject}$ function is sound: suppose e.g. $\mathfrak{subject}(p) = \{t\}$, $\mathfrak{subject}(q) = \{t'\}$, and $\mathfrak{scope}(\alpha) = \{t'\}$. Hence p and $p \wedge (q \vee \neg q)$ do not have the same subject, and $Done_\alpha p \to p$ is an instance of the preservation axiom, while $Done_\alpha(p \wedge (q \vee \neg q)) \to (p \wedge (q \vee \neg q))$ is not. Nevertheless, the latter formula can be deduced from the former standard modal logic principle: as $p \leftrightarrow p \wedge (q \vee \neg q)$ we have $Done_\alpha p \leftrightarrow Done_\alpha(p \wedge (q \vee \neg q))$. Hence $Done_\alpha p \to p$ is equivalent to $Done_\alpha(p \wedge (q \vee \neg q)) \to (p \wedge (q \vee \neg q))$.

We did not formulate such strong compositionality axioms for the $\mathfrak{scope}$ function. The reason is that a speech act might influence more than the topics of its propositional contents. For example, the scope of $\langle \mathsf{Inform}_{u,s}\ \mathit{Class}(1\mathrm{st})\rangle$ contains not only ma_u*:*ma_s*:class* but also ma_u*:*ma_s*:price*. Our hypothesis here is that the scope of a speech act is determined by the subject of its propositional contents together with the integrity constraints e.g. linking destinations, classes, and prices. This is subject of ongoing research.

Last but not least, we note that a possible worlds semantics can be given to our logic by adapting the one presented in [Fariñas del Cerro *et al.*, 1998].

Notes

1 Note that the conversation may be in natural language; however it isn't necessary.

2 We consider that the real world includes everything that is external to the agent (including speaker intention).

3 $Done_\alpha A$ et $Feasible_\alpha A$ are respectively just as $\langle\alpha^{-1}\rangle A$ and $\langle\alpha\rangle A$ of dynamic logic [Harel, 1984].

4 The second term is an abbreviation of the infinite conjunction of Sadek :
$\neg Bel_i Bel_j \neg Bel_i A \wedge \neg Bel_i Bel_j Bel_i \neg Bel_i A \wedge \neg Bel_i Bel_j Bel_i Bel_j \neg Bel_i A \wedge \ldots$

5 The second term is an abbreviation of the infinite conjunction of Sadek :
$\neg Bel_i Bel_j BelIf_j A \wedge \neg Bel_i Bel_j Bel_i BelIf_j A \wedge \neg Bel_i Bel_j Bel_i Bel_j BelIf_j A \wedge \ldots$

6 "Accepting" an act means that we admit that it has been performed.

7 Note that here we differ slightly from Epstein's view. Indeed, Epstein stipulates that $\mathcal{R}(A, A)$ for every formula A. On the contrary, the present axiom makes that $not(\mathcal{R}(\top, \top))$. More generally, we have $\mathcal{R}(A, A)$ iff the set of atoms of A is nonempty. Clearly, this is due to the fact that Epstein does not have the logical operators $\top$ and $\perp$ in his language.

8 Note that an agent might be competent about mental attitudes of some other agent. This means that the former agent controls the latter. We do not exploit this further here.

9 Hence competence should be a 2-argument function. As we only have two participants in our examples, we have dropped the second argument for the sake of simplicity.

References

[Alchourrón *et al.*, 1985] Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50(2):510–530, June 1985.

[Appelt and Konolige, 1989] Douglas Appelt and Kurt Konolige. A nonmonotonic logic for reasoning about speech acts and belief revision. In M. Reinfrank, J. de Kleer, M.L. Ginsberg, and E. Sandewall, editors, *Proceedings of Second International Workshop on Non-Monotonic Reasoning*, volume 346 of *LNAI*, pages 164–175, Allemagne, June 1989. Springer-Verlag.

[Austin, 1962] John L. Austin. *How To Do Things With Words.* Oxford University Press, 1962.

[Büring, 1995] D. Büring. Topic. In P. Bosch and R. v. d. Sandt, editors, *The Focus Book.* 1995.

[Chellas, 1980] B. F. Chellas. *Modal Logic: an introduction.* Cambridge University Press, 1980.

[Cohen and Levesque, 1990a] Philip R. Cohen and Hector J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(2–3):213–261, 1990.

[Cohen and Levesque, 1990b] Philip R. Cohen and Hector J. Levesque. Persistence, intentions, and commitment. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*, chapter 3, pages 33–69. MIT Press, Cambridge, MA, 1990.

[Cohen and Levesque, 1990c] Philip R. Cohen and Hector J. Levesque. Rational interaction as the basis for communication. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*, chapter 12, pages 221–255. MIT Press, Cambridge, MA, 1990.

[Epstein, 1990] R. L. Epstein. *The Semantic Foundations of Logic Volume 1: Propositional Logic.* Kluwer Academic Publishers, 1990.

[Fariñas del Cerro *et al.*, 1998] Luis Fariñas del Cerro, Andreas Herzig, Dominique Longin, and Omar Rifi. Belief reconstruction in cooperative dialogues. In Fausto Giunchiglia, editor, *Proceedings of 8th International Conference on Artificial Intelligence: Methodology, Systems, and Applications (AIMSA'98)*, volume 1480 of *LNAI*, pages 254–266, Bulgaria, September 1998. Springer-Verlag.

[Ginzburg, 1995] J. Ginzburg. Resolving questions i,ii. *Linguistics and Philosophy*, 18:567–609, 1995.

[Goodman, 1961] N. Goodman. About. *Mind*, LXX(277), 1961.

[Grice, 1967] H. Paul Grice. Utterer's meaning and intentions. *Philosophical Review*, 78(2):147–177, April 1967. Also available in [Grice, 1989, Chapter 5].

[Grice, 1975] H. Paul Grice. Logic and conversation. In J. P. Cole and J. L. Morgan, editors, *Syntaxe and Semantics : Speech acts.* Academic Press, New York, 1975. Also available in [Grice, 1989, Chapter 2].

[Grice, 1989] H. Paul Grice. *Studies in the way of words.* Harvard University Press, USA, 3rd edition, 1989.

[Halpern and Moses, 1985] Joseph Y. Halpern and Yoram Moses. A guide to the modal logics of knowledge and belief: preliminary draft. In *Proceedings of IJCAI'85*, pages 480–490, Los Angeles, 1985.

[Harel, 1984] David Harel. Dynamic logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume II. D. Reidel Publishing Company, 1984.

[Herzig, 1998] Andreas Herzig. Logics for belief base updating. In Didier Dubois, Dov Gabbay, Henri Prade, and Philippe Smets, editors, *Handbook of defeasible reasoning and uncertainty management*, volume 3 - Belief Change, pages 189–231. Kluwer Academic Publishers, 1998.

[Hughes and Cresswell, 1972] G.E. Hughes and M.J. Cresswell. *An Introduction to Modal Logic.* Methuen, 2nd edition, 1972.

[Katsuno and Mendelzon, 1992] Hirofumi Katsuno and Alberto O. Mendelzon. On the difference between updating a knowledge base and revising it. In Peter Gärdenfors, editor, *Belief Revision*, number 29 in Cambridge Tracts in Theorical Computer Science, pages 183–203. Cambridge University Press, 1992.

[Lewis, 1972] D.K. Lewis. General semantics. In D. Davidson and G. Harman, editors, *Semantics of natural language.* D. Reidel Publishing Company, 1972.

[Longin, 1999] Dominique Longin. *Interaction rationnelle et évolution des croyances dans le dialogue : une logique basée sur la notion de topique.* PhD thesis, Université Paul Sabatier, Institut de Recherche en Informatique de Toulouse (IRIT), Toulouse, France, to appear 1999.

[McCarthy and Hayes, 1969] J. McCarthy and P. Hayes. Some philisophical problems from the standpoint of artificial intelligence. In B. Meltzer and D. Michie, editors, *Machine Intelligence*, volume 4, pages 463-502. Edinburgh University Press, 1969.

[Perrault, 1990] C. Raymond Perrault. An application of default logic to speech act theory. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*, chapter 9, pages 161–185. MIT Press, Cambridge, MA, 1990.

[Popkorn, 1994] Sally Popkorn. *First Steps in Modal Logic.* Cambridge University Press, Cambridge, 1994.

[Rao and Georgeff, 1991] Anand S. Rao and Michael P. Georgeff. Modeling rational agents within a bdi-architecture. In J. A. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning (KR & R'91)*, pages 473–484, San Mateo, CA, 1991. Morgan Kaufmann Publishers.

[Rao and Georgeff, 1992] Anand S. Rao and Michael P. Georgeff. An abstract architecture for rational agents. In Bernhard Nebel, Charles Rich, and William Swartout, editors, *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR'92)*, pages 439–449, Cambridge, Massachusetts, 1992.

[Rao and Georgeff, 1998] Anand S. Rao and Michael P. Georgeff. Decision procedures for bdi logics. *Journal of Logic and Computation*, 8(3):293–343, June 1998. Special Issue: Computational and Logical Aspects of Multiagent Systems, Oxford univerty Press.

[Reiter, 1980] Ray Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.

[Sadek *et al.*, 1997] David Sadek, Philippe Bretier, and Franck Panaget. Artimis: Natural dialogue meets rational agency. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97).* Morgan Kaufmann Publishers, August 1997.

[Sadek, 1991a] M. D. Sadek. *Attitudes mentales et interaction rationnelle : vers une théorie* *formelle de la communication.* PhD thesis in Computer Sciences, Université de Rennes I, Rennes, France, June 1991.

[Sadek, 1991b] M. D. Sadek. Dialogue acts are rational plans. In *Proceedings of ESCA/ETRW, Workshop on The Structure of Multimodal Dialogue (Venaco II)*, pages 1–29, Maratea, Italie, September 1991.

[Sadek, 1992] M. D. Sadek. A study in the logic of intention. In Bernhard Nebel, Charles Rich, and William Swartout, editors, *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR'92)*, pages 462–473, Cambridge, Massachusetts, October 1992.

[Sadek, 1994] M. D. Sadek. Towards a theory of belief reconstruction: Application to communication. *Speech Communication Journal'94, special issue on Spoken Dialogue*, 15(3–4):251–263, 1994.

[Searle and Vanderveken, 1985] J. R. Searle and D. Vanderveken. *Foundation of illocutionary logic.* Cambridge University Press, 1985.

[Searle, 1969] John R. Searle. *Speech acts: An essay in the philosophy of language.* Cambridge University Press, New York, 1969.

[van Kuppevelt, 1991] J. van Kuppevelt. *Topic en Comment.* PhD thesis, University of Nijmegen, 1991.

[van Kuppevelt, 1995] J. van Kuppevelt. Discourse structure, topicality and questioning. *Linguistics*, 31:109–147, 1995.

Andreas Herzig and Dominique Longin
Institut de Recherche en Informatique de Toulouse (IRIT),
Université Paul Sabatier,
118 Route de Narbonne, F-31062 Toulouse Cedex 4
`mailto: {herzig,longin}@irit.fr`
`http://www.irit.fr/ACTIVITES/EQ_ALG/`

Acknowledgement acts in dialogue openings

Martine Hurault-Plantet, Cecile Balkanski

LIMSI-CNRS

Conversational agents who are collaborating on a task may exchange different types of information regarding this task. They may exchange information that directly relates to the task they are undertaking, such as what actions are to be performed, when and by whom, and what objects of the task are to be manipulated by these actions. They may also exchange information relating to their mutual understanding of what is being said and done, thereby allowing them to establish mutual beliefs about the content of their respective proposals. Finally, agents may also exchange information relating to their acceptance about what is being said and done, thereby allowing them to establish shared agreement about how things are or should be in the way they are undertaking their task.

In this paper, we present a representation of communicative actions providing information of the second type described above, namely, those providing information used to establish mutual beliefs about the content of the speakers' utterances. We refer to these actions as *acknowledgement acts*. This research is part of our development of a collaborative dialogue system that we have implemented for an application simulating a phone desk in an industrial setting. The representation that is presented rests on a study of acknowledgement acts used in dialogue openings of a corpus of man-man dialogues. We show how this representation allows our dialogue system to both understand the user's requests aiming at the establishment of mutual beliefs and answer to them appropriately.

1 Introduction

Conversational agents who are collaborating on a task may exchange different types of information regarding this task. They may, for example, exchange information that directly relates to the task they are undertaking, such as what actions are to be performed, when and by whom, and what objects of the task are to be manipulated by these actions. For instance, agent A may say to agent B: *"I'll write the initial draft for sections 1 and 2 and you take care of sections 3 and 4"*. They may also exchange information relating to their mutual understanding of what is being said and done, thereby allowing them to establish mutual beliefs about the content of their respective proposals. For example, agent B may respond: *"You mean you want me to write up sections 3 and 4?"* to which A may respond *"Yes"*. Finally, agents may also exchange information relating to their acceptance about what is being said and done, thereby allowing them to establish shared agreement about how things are or should be in the way they are undertaking their task[1]. Following up on the preceding example, B could respond with *"okay"*.

Agents need to recognize, in the utterances of their dialogue partners, these different types of information. This recognition allows them to know, at any point in the dialogue, the status of the underlying task, which in turn allows them to determine what to do or say next in order for this task to progress (clarify some ambiguous point, plan further actions, or reach agreement on the value to assign to an action parameter, for instance). A major difficulty in recognizing these different communication acts resides in the fact that the information conveyed is not always explicit. In particular, information concerning mutual understanding or mutual agreement is often expressed implicitly. This happens when one agent A makes a proposal P_A and the other agent B continues with a proposal P_B, coherent with P_A. By proceeding with a new proposal, B is implicitly signaling both his understanding and his acceptance of proposal P_A. If, on the other hand, B utters *"okay"*, then there is ambiguity on whether he is explicitly signaling his acceptance of A's proposal (and therefore his understanding) or only his understanding (this ambiguity may be resolved by B's subsequent utterance).

In this paper, we are concerned with communicative actions providing information of the second type described above, namely, those providing information used to establish mutual beliefs about the content of the speakers' utterances. Following Traum (1994), we refer to these actions as *acknowledgement acts*. This research is part of our development of a collaborative dialogue model that we have implemented for an application simulating a phone desk in an industrial setting (Balkanski and Hurault-Plantet, 1997, 1999, Hurault-Plantet and Balkanski 1998). Here we focus on the representation of acknowledgement acts, with the aim of allowing our system to both understand the user's requests aiming at the establishment of mutual beliefs and answer to them appropriately.

We became interested in dialogue openings because, as shown in the corpus (Castaing 1993) which we used to build the knowledge base of our system, these openings may contain complex exchanges aimed at establishing mutual beliefs concerning the successful establishment of the communication. The following dialogue[2] illustrates the type of opening found in this corpus and presents an insightful example of the agents' efforts to explicitly establish these mutual beliefs. Although the process of opening dialogues appears trivial, the resulting dialogues are rich in variation, as we will show in the next section of this paper. In the example of Figure 1, after the operator's initial utterance, in (2), the caller first makes sure that the communication is well established (3), which the operator confirms (4), then he verifies that he is talking to someone at the right organization (5), which the operator again confirms (6), and then, only after these preliminary exchanges, does he state his goal, which is to talk to a particular person (7).

(1)	L: < appel >	Caller:	< phone call >
(2)	S: CNRS bonjour	Operator:	hello this is the CNRS
(3)	L: allô oui	C:	hello
(4)	S: oui	O:	yes
(5)	L: c'est le CNRS	C:	I'm at the CNRS
(6)	S: oui monsieur	O:	yes sir.
(7)	L: je voudrais madame Boissy au poste 26-57	C:	I'd like Mrs Boissy extension 26-57

Figure 1. Dialogue N°1-A(1)N11

The remaining of this paper is organized as follows. Section 2 highlights the main aspects of our corpus study, focusing on utterances of dialogue openings which serve to establish mutual beliefs between the operator and the caller. Section 3 describes the representation of the acknowledgement acts that are part of our dialogue model, and provides a brief description of our implemented dialogue system. In section 4, we present a simulation of our system's processing of a sample dialogue. We show in particular how acknowledgement acts are used by the reasoning processes of our model, both during the interpretation of the caller's utterances and the generation of the system's utterances. We conclude in section 5 with a discussion of related work and future research.

2 Corpus study: dialogue openings

Our data is drawn from a corpus of human-human dialogues, selected and transcribed from a day-long recording of phone calls at a phone desk in an industrial setting (Castaing 1993). The full recording consists of about 500 dialogues, 353 of which have been transcribed. These dialogues are classified as request dialogues (275) or as dialogues resuming a previous request (78), this second type occurring when the operator returns to a caller after a certain lapse of time, when the line is busy or is not answered. In the present analysis, we focus on the initial request dialogues, leaving the second group of dialogues for a future study.

2.1 Main patterns

As illustrated in Figure 1, request dialogues begin by the caller's phone call, followed by a response by the operator. The operator's response consists most often of the name of organization followed by a greeting (see Table 1), or, less frequently, of just the name of the organization. The remaining cases include *"allô"*[3], *"bonjour" ("hello")*, *"oui" ("yes")*, and combinations of these. Despite the fact that *"bonjour"* is often used as a greeting in ordinary conversation, the variation in the possible responses seems to indicate that the operator's underlying goal is not only to greet the person who called, but primarily to accept the communication. Schegloff (1968) reached the same conclusion in his study of dialogue openings. He analyses the phone call and the first utterance of the respondent as constituting a "summons/answer" sequence, very similar to the question-response sequence. We consider, in addition, that the calling and answering action sequence allows the conversational partners to perform another (higher-level) action, namely the one which consists in establishing the communication between the agents of these two actions. This aspect of dialogue openings will be formalized in the next section.

Acceptance of the communication on the part of the operator	**Percentages of dialogues including this acceptance**
"CNRS bonjour" (CNRS hello)	65%
"CNRS"	23%
"allô CNRS", "oui CNRS", "allô CNRS bonjour", "bonjour", "allô bonjour", "oui bonjour", "oui", "allô".	12%

Table 1 : Frequency of the different types of communication acceptances used by the operator

After this initial exchange, the dialogue openings of our corpus present a number of variations which we have divided into five main patterns as presented in Table 2. In the dialogues, C is the caller and O, the operator. Acts between slashes (//) indicate an overlap between the two speakers' corresponding utterances.

Establishment of the communication: (1) C: <phone call> (2) O: <communication acceptance>		**Percentages of dialogues including this pattern**
Pattern 1	(3) C : <question>	6,5%
Pattern 2	(3a) C : <acknowledgement> (3b) <question>	54%
Pattern 3	(3a) C : <acknowledgement> (3b) O : // <acknowledgement> // (3c) C : <question>	18%
Pattern 4	(3) C : <request for acknowledgement> (4) O : <acknowledgement> (5) C : <question>	6,5%
Pattern 5	(3) C : <request for acknowledgement> (4) O : <acknowledgement> (5a) C : <acknowledgement> (5b) <question>	8%
Other patterns		7%

Table 2 : Dialogue opening patterns

These patterns display regularities that we observed in the dialogue openings of our corpus. Their formalization rests on an interpretation of the dialogue participants' exchanges in terms of the types of communicative acts that are used, such as acknowledgements and requests for acknowledgement. We detail these patterns in the following sections, presenting for each one of them the different ways in which these communication acts are expressed.

2.2 Pattern 1

(1) C: <phone call>
(2) O: <communication acceptance>
(3) C : <question>

Example of pattern 1:

(1) L: < appel >	(1) C : <phone call>
(2) S: CNRS	(2) O : CNRS
(3) L: je voudrais le poste 25-42 Mr L	(3) C : I would like extension 25-42 Mr.L

Figure 2 : Dialogue N°2-A(3)N40

This first pattern is the simplest. It consists of the two actions that are necessary for establishing the communication between the two speakers, as described above, which are immediately followed by the request of the speaker. This pattern may contain no greeting, as is the case in the example of Figure 2.

2.3 Pattern 2

(1) C: <phone call>
(2) O: <communication acceptance>
(3a) C: <acknowledgement>
(3b) <question>

Example of pattern 2:

(1) L: < appel >	(1) C : <phone call>
(2) S: CNRS	(2) O : CNRS
(3a) L: bonjour	(3a) C : Hello
(3b) le poste 22-69 s'il vous plait	(3b) extension 22-69 please

Figure 3 : Dialogue N°2-A(3)N52

The second pattern is the most common. It represents a little more than half the dialogue openings of our corpus. In these dialogues, the caller does not immediately state his goal (<question> in (3b)), but first answers *"bonjour" ("hello")* or *"bonjour madame"*[4], in 48% of the cases, *"oui bonjour" ("yes hello")* or *"oui bonjour Madame"*, in 22% of the cases, *"oui" ("yes")* in 19% of the cases, and *"allô"* or combinations of these different expressions in the remaining cases (see Table 3).

Acknowledgement on the part of the caller		**Percentages of dialogues of pattern 2 including this acknowledgement**
"bonjour" "bonjour Madame"	(hello)	48%
"oui"	(yes)	19%
"oui bonjour" "oui bonjour Madame"	(yes hello)	22%
"oui Madame", "allô", "allô bonjour", "allô bonjour Madame", "bonsoir Madame", "bonsoir", "Madame"		11%

Table 3: Caller acknowledgement acts in pattern 2

Following up on our discussion in the introduction (section 1), we label these expressions as acknowledgements, since they appear to be used as confirmations, on the part of the caller, that the communication has been established. Although the expression *"bonjour"* is ambiguous, our corpus study supports its interpretation as an acknowledgement in this context. Indeed, utterances (2) and (3a) do not constitute a simple exchange of greetings since the corpus shows that a *"hello"* on the part of the operator is not always followed by a *"hello"* on the part of the caller, and, inversely, a *"hello"* on the part of the caller is not necessarily preceded by a *"hello"* on the part of the operator. The sample dialogue opening in Figure 3 illustrates this point.

2.4 Pattern 3

(1) C: <phone call>
(2) O: <communication acceptance>
(3a) C: <acknowledgement>
(3b) O: // <acknowledgement> //
(3c) C: <question>

Example of pattern 3:

(1) L : < appel >	(1) C : <phone call>
(2) S : CNRS	(2) O : CNRS
(3a) L : bonjour	(3a) C : hello
(3b) S : // bonjour //	(3b) O : // hello //
(3c) L : pouvez-vous me passer le CDI	(3c) C : could you transfer me over to the CDI

Figure 4 : Dialogue N°2-A(3)N38

In this third pattern, the operator acknowledges (3b) the *"hello"* of the caller (3a), while the caller continues speaking (3c). As a result, the operator's greeting partially overlaps the caller's question. In comparison to the second pattern, there is also less variation in the range of acknowledgment used. Here, with very few exceptions, there is a *"hello"* greeting on the part of the caller followed by a similar greeting on the part of the operator, as shown in tables 4 and 5.

Acknowledgement on the part of the user	Percentages of dialogues including this acknowledgement
"bonjour" (hello) "bonjour Madame"	42%
"oui bonjour" (yes hello) "oui bonjour Madame"	46%
"allô bonjour" "allô"	12%

Table 4 : Caller acknowledgement acts in pattern 3

Acknowledgement on the part of the operator	Percentages of dialogues including this acknowledgement
"bonjour" (hello)	94%
"bonjour Madame" "bonjour Monsieur" "oui"	6%

Table 5 : Operator acknowledgement acts in pattern 3

We interpret this exchange of greetings as a double confirmation (acknowledgement) that the communication has been successfully established between the two parties. Each confirmation (acknowledgement) reinforces the mutual belief that this communication has been established and that each speaker is ready to listen to the other. The overlap between the greeting of the operator and the question of the caller indicates that the caller was not expecting a response to his own greeting. The operator interprets this *"hello"* as a request, since she answers to it, while the caller continues his turn by his question, indicating that for him this *"hello"* was just an acknowledgement. It therefore appears that, in this third pattern, there is a certain confusion in the exchange of acknowledgements, as if the caller placed himself in the second pattern while the operator placed herself in the fourth pattern, which is presented next.

2.5 Pattern 4

(1) C: <phone call>
(2) O: <communication acceptance>
(3) C: <request for acknowledgement>
(4) O: <acknowledgement>
(5) C: <question>

Example of pattern 4:	
(1) L: < appel >	(1) C : <phone call>
(2) S: CNRS bonjour	(2) O : CNRS hello
(3) L: allô	(3) C : hello
(4) S: oui	(4) O: yes
(5) L: j'aurais voulu parler à Mr C s'il vous plait	(5) C: I would have liked to speak to Mr.C please

Figure 5 : Dialogue N°1-B(2)N54

The main difference between this pattern and the preceding one is in the lapse of time that occurs after the caller's first utterance in (3). Here there is no overlap between the operator's response (4) and the caller's question (5). In this context, utterance (3), *"allô"*, is, therefore, a request. Contrary to the French *"bonjour"*, the expression *"allô"*, which is used a number of times in this pattern (see table 6), is not a greeting. It can be interpreted as an acceptance of the communication, when it is used in response to a phone call, and as a request otherwise[5]. This second case occurs, for instance, when the callee picks up the phone but does not answer. The caller then utters an interrogative *"allô"* to know if the callee is ready to listen. Schegloff (1968) analyses this utterance as a repetition of the request for communication, initially performed by the phone call. This request is the first action of the "summons/answer" sequence of dialogue openings. It apparently failed since the callee picked up the phone but did not indicate that he was ready to communicated. Therefore, the caller repeats it, but by executing it differently.

Request for acknowledgement on the part of the caller		**Percentages of dialogues including this request**
"bonjour" "bonjour Madame"	(hello)	47%
"allô" "allô CNRS"	(hello)	35%
"oui bonjour" "allô bonjour"	(yes hello)	18%

Table 6 : Caller acknowledgement acts in pattern 4

In pattern 4, we interpret the caller's *"allô"* as a different kind of request, namely a request for acknowledgement of the establishment of the communication. Since the operator answered, we may assume that communication has been successfully established. This request is followed by an acknowledgement on the part of the operator, expressed as a greeting (*"bonjour"*) or an acceptance (*"oui"*). The operator acknowledgement in this fourth pattern is more often expressed as an acceptance (*"oui"*) than is the case for the caller acknowledgement in the second pattern (compare tables 3 and 7). This may be explained by the fact that an acknowledgement which follows a request for acknowledgement (as in pattern 4) is better expressed as a confirmation.

Acknowledgement on the part of the operator	Percentages of dialogues including this acknowledgement
"bonjour" (hello) "bonjour Madame"	53%
"oui"	47%

Table 7 : Operator acknowledgement acts in pattern 4

2.6 Pattern 5

(1) C: <phone call>
(2) O: <communication acceptance>
(3) C: <request for acknowledgement>
(4) O: <acknowledgement>
(5a) C: <acknowledgement>
(5b) <question>

Example of the Pattern 5:

(1) L: < appel >	(1) C : <phone call>
(2) S: CNRS bonjour	(2) O : CNRS hello
(3) L: allô	(3) C : hello
(4) S: oui	(4) O: yes
(5a) L: oui bonjour	(5a) C: yes hello
(5b) j'aurais voulu parler à Mr T s'il vous plait	(5b) I would have liked to speak to Mr.C please

Figure 6 : Dialogue N°1-B(2)N53

The main difference between this pattern and the preceding one is in the additional acknowledgement on the part of the caller, in (5a), preceding his question. The requests for acknowledgement on the part of the caller, the acknowledgements on the part of the operator, and the acknowledgement on the part of the caller as a response to the acknowledgement on the part of the operator, are expressed in different ways, as described in tables 8, 9 and 10.

In this fifth pattern, the sequence consisting of the caller's *"allô"* followed by the operator's *"oui"* (*"yes"*) is by far the most frequent. Furthermore, as shown in table 8, this initial acknowledgement exchange involves not only an acknowledgement of the successful establishment of the communication, but also an acknowledgement of the right destination of the call. The caller then reinforces the mutual belief by a further acknowledgement, illustrated in (5a). This acknowledgement is similar to the one used in the second pattern (see (3a) in Figure (3)), and displays the same kind of variations (compare tables 10 and 3).

Request for acknowledgement on the part of the caller	Percentages of dialogues including this request
"allô" (hello)	71%
"allô CNRS" (hello CNRS) "oui c'est le CNRS" "allô oui bonjour CNRS"	29%

Table 8 : Caller acknowledgement acts in pattern 5

Acknowledgement on the part of the operator	Percentages of dialogues including this acknowledgement
"oui" (yes)	90%
"allô bonjour" (hello) "bonjour"	10%

Table 9 : Operator acknowledgement acts in pattern 5

Acknowledgement on the part of the user		Percentages of dialogues including this acknowledgement
"oui bonjour" "oui bonjour Madame"	(yes hello)	43%
"bonjour Madame" "bonjour"	(hello)	43%
"oui" "allô"	(yes)	14%

Table 10 : Caller acknowledgement acts in response to operator acknowledgement acts in pattern 5

2.7 Other patterns

A number of dialogues cannot be categorized by one of the above patterns. These patterns display the complexities characteristic of pattern 5, with a request for confirmation concerning the name of the company, as illustrated in Figure 7 (as well as in Figure 1, in the Introduction), or additional confirmations relating to the successful establishment of the communication, as illustrated in Figure 8.

(1) L: < appel >	(1) C : <phone call>
(2) S: CNRS	(2) O : CNRS
(3) L: allô	(3) C : hello
(4) S: oui	(4) O: yes
(5a) L: CNRS Bellevue	(5a) C : CNRS Bellevue
(5b) S : // oui //	(5b) O : // yes //
(5c) L: je voudrais parler à Mme L s'il vous plait	(5c) C : I'd like to speak to Mrs L please

Figure 7 : Dialogue N°2-A(3)N56

(1) L: < appel >	(1) C : <phone call>
(2) S: CNRS	(2) O : CNRS
(3) L: allô	(3) C : hello
(4) S: oui	(4) O: yes
(5a) L: Bonjour Madame	(5a) C : hello mrs
(5b) S : // bonjour Monsieur //	(5b) O : // hello sir //
(5c) L: est-ce que je peux parler à Mme L s'il vous plait	(5c) C : may I speak to Mrs L please

Figure 8 : Dialogue N°2-A(3)N80

3 Acknowledgement acts in the dialogue model

In this section, we describe the action and recipe structures of our dialogue model, focusing on the representation of acknowledgement acts. We then provide a brief overview of the different components of the architecture of our system. This overview provides background information that is necessary for understanding the simulation of the system's processing of a sample dialogue opening that is presented in the following section.

3.1 Acknowledgement actions and recipes

Our dialogue model was designed with the aim of modeling an agent capable of collaborating with a human agent by participating in a task-oriented dialogue. The mental state of the agent being modeled is represented by a set of beliefs and intentions about communicative and domain actions, relations among them, and entities that can be used as action parameters.

Some of these beliefs are the agent's "permanent" beliefs concerning the actions, recipes, and entities that make up the agent's knowledge base[6]. Actions may be basic or complex. While basic actions are performable at will, under certain conditions, complex actions have associated recipes. Recipes are composed of a set of constituent acts and associated constraints, necessary for the performance of a complex act. Each constituent act contributes to realizing the complex act. A sample recipe is given below in Figure 9. As shown in this figure, we define recipes as two-level trees, the root being a complex action, the leaves being the constituent actions, basic or complex, necessary for the execution of the head action. The recipe node, represented by an oval box, contains constraints on parameters which have to be satisfied to allow for coherence among the parameters (agents, time, entities) of the actions in the recipe, as well as temporal constraints on the execution of constituent actions and, when necessary, other contextual constraints.

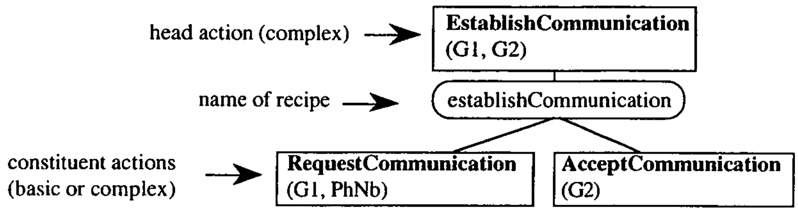


Figure 9. establishCommunication recipe

The recipe defined in Figure 9 is the establishCommunication recipe, which captures the relation, mentioned in the preceding section, between the initial calling and answering actions and the (higher-level) action, establishing communication, the performance of which results from the performance of the two constituents actions.

We focus in this paper on explicit acknowledgement actions, with the aim of allowing the agent being modeled to understand requests for acknowledgement and to answer appropriately to them. We therefore introduce two recipes for obtaining an acknowledgement. These recipes are presented in Figure 10, and both allow an agent G1 to obtain on the part of his or her conversational partner G2 an acknowledgement concerning a proposition P.

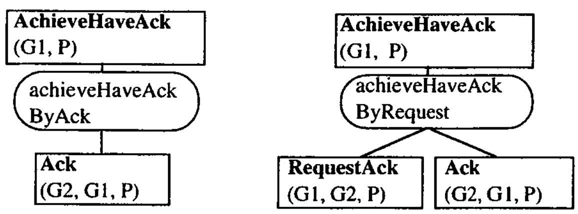


Figure 10. Acknowledgement actions and recipes

The first recipe, achieveHaveAckByAck allows G1 to obtain an acknowledgement by means of G2 performing the communicative action of performing this acknowledgement. The second recipe, achieveHaveAckByRequest allows G1 to obtain an acknowledgement regarding P by means of G1 performing the communicative action of requesting this acknowledgement, and then by G2 performing the communicative action of performing the acknowledgement.

The agent being modeled also has beliefs about the current dialogue context. These include the system's beliefs about the system's and the user's mutual beliefs regarding the task they are collaborating on, as well as about the actions that they intend to perform. These are temporary beliefs which last, at most, the time of a dialogue. The acknowledgement acts presented above allow, in particular, to establish such mutual beliefs as the dialogue progresses.

3.2 Dialogue system

The theoretical framework of our dialogue system is based on the theory of discourse developed by Grosz and Sidner (1986) and the theory of Shared Plans developed by Grosz and Sidner (1990), Grosz and Kraus (1996) and further extended by Lochbaum (1998). Shared Plans were introduced to model the set of beliefs and intentions about actions to be performed that agents must hold for their collaboration to be successful. Agents begin with a partial Shared Plan and gradually update it as the dialogue progresses. The definition of a Shared Plan therefore provides an important context for interpreting utterances, each utterance being interpreted as providing information that is necessary for the establishment of a full SharedPlan.

Lochbaum (1998) subsequently proposed a computational model, based on the SharedPlan formalism, for recognizing intentional structure and utilizing that structure in discourse processing. This model includes a dynamic structure, named a recipe graph, or Rgraph, which results from instantiating and composing recipes as the dialogue, and therefore the underlying task, progress. The Rgraph corresponds to a partial representation of the agents' Shared Plan, and represents the beliefs of the agent being modeled as to how all of the acts underlying the agents' discourse are related at a given point in the dialogue. These are acts that have been, or will have to be, performed by the collaborating agents to accomplish their individual and shared objectives. Rgraphs include instantiations of the parameters of the actions present in the graph, as well as status information, indicating, for each action in the Rgraph, the belief of the agent being modeled about its state in the agents' underlying plan (and in particular its order of execution within the plan).

Our computational model extends Lochbaum's interpretation algorithm to allow for the treatment of a wider range of dialogues, and introduces an algorithm of task advancement which guides the generation process and allows for the integration of planning and acting, thereby facilitating cooperation among agents. The goal of this algorithm is to determine which actions the system can perform, and in which order, so as to allow the overall task to progress. It also determines when to plan further actions, when to replan an action that failed, or when to produce an utterance and what it should contain. It does so by manipulating the Rgraph, making choices by assigning to the various options priorities guided toward action execution.

The overall structure of the Task Advancement algorithm is a loop. Its initialization consists in determining the current action in the Rgraph (namely the first action to be performed in the temporal order of the actions in the Rgraph, or, if all actions have been performed, the last action performed), its stopping condition is verified when the system needs to produce a speech act, and its body consists in applying a number of rules to allow the task to progress. The choice of the rule to apply depends of the status of the action in the Rgraph. Status information evolves as the dialogue proceeds, and indicates, for each action in the Rgraph, the belief of the agent being modeled about its state in the agent's underlying plan, e.g., if it is only intended on the part of the agent, or has also been planned for, or has already been executed.

A schematic representation of the architecture of our system is given in Figure 11. In the phone desk domain, we model two agents: the user, who assumes the role of the caller, and the system, which assumes the role of the operator.

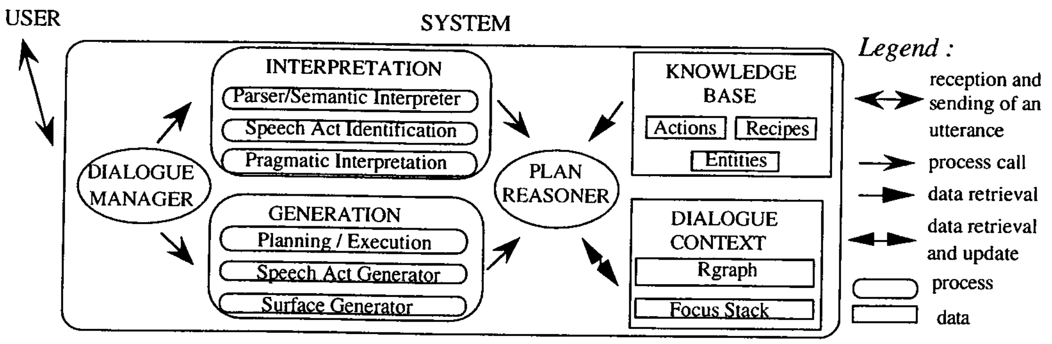


Figure 11. Schematic diagram of the system architecture

The system's knowledge base consists of the system's permanent beliefs about actions, recipes and entities. These actions were defined on the basis of our corpus analysis, and may be task related (e.g., EstablishCommunication in Figure 9) or task independent (e.g., AchieveHaveAck in Figure 10).The dialogue context includes a recipe graph and a focus stack, and represents the beliefs of the system concerning the dialogue.

The system's reasoning capabilities are built into four main components, a Dialogue Manager, an Interpretation module, a Generation module and a Plan Reasoner, as shown in Figure 11. The Dialogue Manager receives the input, a natural language utterance, sends it to the Interpretation module, then calls the Generation module to produce its response.

The Interpretation module's role is to update the dialogue context so as to reflect the contribution of the current utterance to the dialogue. This module is divided into three submodules. The Parser and Semantic Interpreter parses sentences input by the user, and translates them into a semantic representation. The role of the Speech Act Identification module is to recognize from this semantic representation the speech act (e.g., suggest) as well as the act that is related to the task and which will be reasoned about. Finally, the Pragmatic Interpretation module determines the meaning of the action underlying the utterance in the current dialogue context, and updates the dialogue context accordingly. To do so, this module calls upon the Plan Reasoner, whose role during interpretation is to find a way of integrating the act A underlying the current utterance to the Rgraph. These reasoning processes are modeled by the Interpretation algorithm.

The goal of the Generation module is to make as much progress as possible in the task underlying the dialogue, in reaction to the interpretation of an utterance, and then to generate the system's response. Different submodules are again introduced for this purpose, which mirror the three submodules of the Interpretation module. The Planning/Execution module, calling upon the Plan Reasoner, tries to further build the Rgraph, executing actions when necessary, and then plans the content of the system's response. These reasoning processes are modeled by the Task Advancement algorithm described earlier. The Speech Act Generator and Surface Generator modules then further structure the system's utterance.

Given our current research interests, we have focused on the Plan Reasoner, temporarily leaving aside the first and last two submodules. Details about the Interpretation and Task Advancement algorithms that are part of this module can be found elsewhere (Balkanski and Hurault-Plantet, 1999). We have implemented, in Smalltalk, an application that integrates our dialogue model and that simulates a receptionist for phone calls at a phone desk. The application is not connected to a real telephone system, but allows the user to simulate a phone call by clicking a button, and to hang up by clicking another button. The user types his "utterance" in a text window. The system interprets this utterance, and responds by displaying its utterance in another window. The Parser/Semantic Interpreter and Speech Act Identification, as well as the Speech Act Generator and Surface Generator modules are still very preliminary. On the other hand, the Pragmatic Interpretation and Planning/Execution modules, and the associated Interpretation and Task Advancement algorithms that are part of the Plan Reasoner are fully implemented.

Our application currently contains 14 actions and 10 recipes, which is sufficient for handling the request dialogues of our corpus which are based on an extension (e.g., *"Extension 22-69 please"*, in Figure 3) or the name of a person (e.g., *"I would like to speak to Mr.C, please"*, in Figure 5). This type of dialogue makes up about 60% of our corpus. The remaining dialogues include essentially requests for communication with a particular department (e.g., *"I'd like the Personnel Department")*, requests for information that lead to the operator transferring the caller to a particular department, as well as subdialogues resuming a previous request and occurring after a certain lapse of time, when the line is busy or is not answered (called "dialogues de reprise" in the original corpus).

4 Sample dialogue opening

In this section we simulate the system's processing of the following sample dialogue opening, similar to the one given in Figure 5, and belonging to pattern 4.

(1)	User:	<phone call>	
(2)	System:	CNRS bonjour.	(CNRS hello)
(3)	User:	allô.	(hello)
(4)	System:	oui.	(yes)
(5)	User:	je voudrais	(I'd like ...)

Figure 12. Sample dialogue opening

For the system, the task starts out with the phone call made by the user, indicated in (1), and simulated by the user clicking a button in our interface. The system's Dialogue Manager receives the call. This manipulation act is directly translated into the RequestCommunication(User, System) action, and sent to the pragmatic Interpretation module, which in turn sends it to the Plan Reasoner to update the Rgraph on the basis of this action. Since the Rgraph is initially empty, the Plan Reasoner's Interpretation algorithm uses different heuristic rules to handle initial utterances. One of these rules states that if the user executes an action A, it may be with the goal of executing another action B to which A contributes. In this case, the Plan Reasoner searches the database for recipes containing the action that the user just executed. Here, the Plan Reasoner finds that the RequestCommunication action is part of the establishCommunication recipe as well as the transferPerson recipe. Both recipes are defined in Figure 13. The transferPerson recipe is not used because of the constraint stating that G1 has to be employed by the company. In this particular context, G1 would be the user (agent of the RequestCommunication action) and the constraint would therefore not be satisfied.

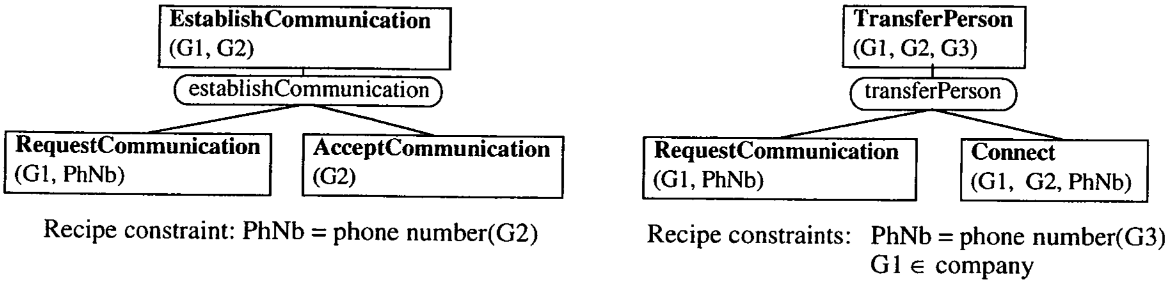


Figure 13. establishCommunication and transferPerson recipes

Thus the Plan Reasoner initializes the Recipe Graph with the establishCommunication recipe and associated actions, with instantiated parameters, as shown in Figure 14. The Plan Reasoner may thus infer implicit goals, and the Recipe Graph represents the system's current beliefs about the user's intention. As shown in Figure 14, the status information associated to each action in the Rgraph shows that the system believes that the EstablishCommunication action has been planned for, that the RequestCommunication action has been performed and that the AcceptCommunication action is intended.

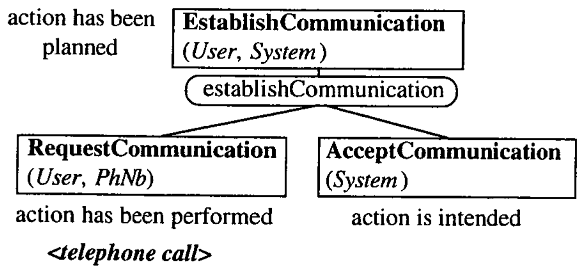


Figure 14. Recipe graph after the phone call in (1)

The pragmatic Interpretation module returns a success signal to the Dialogue Manager, which then calls upon the Planning module. This module in turns calls the Plan Reasoner which applies the Task Advancement algorithm to try to advance the task. The Plan Reasoner looks for the first action to be executed (initialization of the Task Advancement algorithm) by temporal order. It finds the AcceptCommunication action, a communicative action for which the system is the agent (stopping condition of the Task Advancement algorithm). The Planning module thus updates its status to "action performed" and sends it to the generation modules which then produce the standard greeting (2): "CNRS hello".

This utterance corresponds to the performance of the AcceptCommunication(*System*) action. This performance results in that of the EstablishCommunication(*User*, *System*) action. The status information of the Rgraph is therefore updated as indicated in Figure 15.

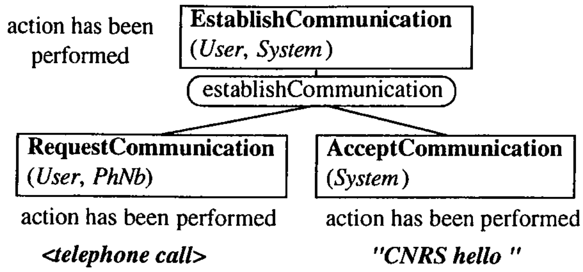


Figure 15. Recipe graph after the generation of utterance (2)

The Dialogue Manager then receives the user's utterance:

(3) User: allô.

The Plan Reasoner interprets the action underlying this utterance in the dialogue context represented by the Rgraph in Figure 15. As shown in section 2.5 of our corpus study, this communicative action has a several possible interpretation. It can be interpreted as an acceptance of the communication when used as an answer to a phone call, or a reexecution of a request for communication when preceded by an acceptance action which failed, or a request for acknowledgement, or also (but more rarely) an acknowledgement.

Since the last action performed in the Rgraph of Figure 15 is the EstablishCommunication(*User*, *System*) action and not a RequestCommunication action, the communicative action "allô" being interpreted cannot be interpreted as an acceptance. It cannot be interpreted either as a reexecution of the RequestCommunication action because the AcceptCommunication has been successfully executed. It is not an acknowledgement either because the user did not follow up on his utterance. The user's communicative action is therefore interpreted as a request for acknowledgement. This action is part of the achieveHaveAck recipe given in Figure 10. As shown in this recipe, a request for communication is performed with the aim of obtaining an acknowledgement.

The Plan Reasoner instantiates the actions of this recipe, and creates an acknowledgement subgraph to be associated with the EstablishCommunication(*User*, *System*), as shown in Figure 16.

The link between the two subgraphs represents a background link. The other links in a Rgraph are used to represent relations between actions that contribute to the realization of the head action of the graph (by generation, enablement, or a combination of these, Balkanski and Hurault-Plantet, 1997). Acknowledgement actions, on the other hand, do not contribute to the performance of an action of the Rgraph, but to the establishment of a mutual belief concerning such an action. Therefore, they are not added directly to the Rgraph, but as "background" information, connected to the actions they relate to.

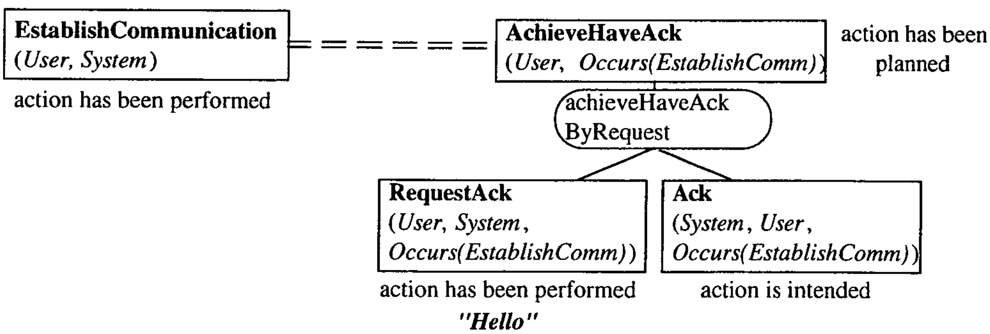


Figure 16. Acknowledgement graph after the interpretation of utterance (3)

Having interpreted the input utterance, the Plan Reasoner now applies the Task Advancement algorithm. The action in focus is EstablishCommunication(*User*, *System*). Since it has already been performed, the Plan Reasoner checks if an acknowledgement graph is associated to this action, which it is. The Plan Reasoner then applies the Task Advancement algorithm to this graph. The next action to be performed is Ack(*System*, *User*, *Occurs(EstablishComm)*). This is a communicative action whose agent is the system. This situation is a stopping condition of the Task Advancement algorithm, and the Planning module therefore sends the action to the generation modules which produce utterance (4), "yes". Finally, the status information for this action is updated to "action performed".

5 Conclusion

Dialogue openings have not been studied much, maybe because they are considered secondary and that "real" dialogue issues begin with the statement of the speaker's goal. As a result, most dialogues models are designed to be used for processing dialogues that begin with the statement of the speaker's goal. However, we claim that not only do dialogue openings constitute a topic of research in their own right (they occur in "real" dialogues, often, and therefore deserved to be understood), but that the proposals made to handle them are relevant and useful for processing the rest of the dialogue. In particular, the acknowledgement and request for acknowledgement acts proposed in this paper for processing dialogue openings can be used for processing the numerous instances of acknowledgement related exchanges that occur in the course of a dialogue.

There is, however, some related literature, and in particular Schegloff's study of conversational openings (1968). He proposes to model openings by means of the "summons/answer" sequence which was referred to earlier in this paper. Our formalization extends this approach by assigning a goal to this sequence, thereby allowing to represent it as a recipe, and to use it in a plan. Our work also focuses on the establishment of mutual beliefs, an area of research that he does not examine.

Cherny (1995) analyzes conversational openings and closings in a sociological study of speech acts in MUDs (Multi-User-Dimensions in text-based virtual realities). She views these dialogue sequences as conventional actions, conventions which contribute to structuring interactions, but that are not always strictly adhered to. Dialogue openings and closings therefore appear to be both necessary and open to a certain level of variation. However, even if dialogue openings relate to conventions, these conventions are used with a certain goal, and are therefore subject to the same type of reasoning processes that are used in the rest of the dialogue. The dialogue openings in our corpus where also shown to be subject to variation, while being centered around goals that are identifiable, such as establishing communication, and assuring one another of the correct mutual understanding of the content of one's utterances. If these goals are not identified, it is not possible to respond appropriately to utterances such as those that appear in dialogue openings and that were identified in our corpus study.

Traum (1994) develops a computational theory of grounding acts, which, when implemented within a dialogue system, can be used to determine, for any given state of the dialogue, whether material has been grounded or what it would take to ground that material. This work provided a theoretical context that contributed to the definition of the representation of acknowledgement acts in our model, a representation allowing the system to identify the user's goal of establishing a mutual belief and to answer to it.

In this paper, we showed how to model explicit acknowledgement acts, and how they are used to establish mutual beliefs. This modeling is necessary to allow our system to identify the user's requests for acknowledgement, and to respond to them appropriately. The goal of our current research is now to link the treatment of these explicit acknowledgement acts to the inferences made by the system to establish mutual beliefs on the basis of implicit acknowledgement acts.

References

Balkanski, C., Hurault-Plantet, M. (1997). Communicative Actions in a Dialogue Model for Cooperative Discourse: an initial report. In *proceedings of AAAI Fall 1997 Symposium on communicative action in humans and machines*. Cambridge, MA.

Balkanski, C., Hurault-Plantet, M. (1999). Cooperative requests and replies in a collaborative dialogue model. To appear in the *IJHCS Special Issue on Collaboration, cooperation and conflict in dialogue systems.*

Castaing, M.F. (1993). Corpus de dialogues enregistrés dans un standard téléphonique. *Notes et documents LIMSI 93-9.* LIMSI/CNRS, Orsay, France.

Cherny, L., (1995). The modal complexity of Speech Events in a Social MUD. Electronic Journal of Communication, 5(4).

Grosz, B., Sidner, C. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics,* 12(3):175-204.

Grosz, B., Sidner, C. (1990). Plans for discourse. In P.R.Cohen, J.L.Morgan, and M.E.Pollack, eds. *Intentions and Communication.*, pp.417-444, MIT press, Cambridge, MA.

Grosz, B., Kraus, S. (1996). Collaborative plans for complex group action. *Artificial Intelligence* 86(2).

Hurault-Plantet, M., Balkanski, C. (1998). Communication and manipulation acts in a collaborative dialogue model. In *proceedings of the Second International Conference on Cooperative Multimodal Communication,* Tilburg, Nederlands.

Lochbaum, K. (1998). A Collaborative Planning Model of Intentional Structure. *Computational Linguistics,* 24(4):525-572.

Schegloff, E., (1968). Sequencing in Conversational Openings. *American Anthropologist,* 70(6):1075-1095.

Traum, D., (1994). A Computational Theory of Grounding in Natural Language Conversation. *TR-545, Ph.D. dissertation.* Department of Computer Science, University of Rochester.

Martine Hurault-Plantet and Cecile Balkanski
LIMSI/CNRS, B.P. 133, 91403 Orsay, France

---

[1] Using Traum's terminology (1994), communicative acts concerning information of the second type may be described as *grounding* acts, while those concerning information of the third type may be described as *acceptance* acts.
[2] In the corpus all names cited are fictitious. Extensions are not because the phone desk has been reorganized and the numbers no longer exist.
[3] The word *"allô"*, used exclusively in phone calls, will be commented on later in this paper.
[4] This expression, very common in French, is awkward when literally translated into English.
[5] Sometimes, but much more rarely, it can be interpreted as an acknowledgement, as shown in Table 3.
[6] By permanent, we mean stable over a large number of dialogues. These are beliefs that are not revisable during a dialogue.

Speech Acts and Dialogue Control in Task-Oriented Spontaneous Dialogue

MASAHITO KAWAMORI

NTT Laboratories

AKIRA SHIMAZU

Japan Advanced Institute of Science and Technology

Abstract

In a spontaneously spoken dialogue, speech acts are not necessarily expressed in full-fledged well-formed sentences; they are often couched in rather fragmentary, truncated semi-sentences or phrases. We show that a naturally occuring spontaneous dialogue is full of features that are beyond the traditional analyses of speech acts, features such as incomplete sentences, repetitions, repairs, backchannels, and other interejections. These features are often closely related to the system of turn-taking or control passing. Discourse representations based on DRT are employed to represent spontaneous dialogues in an incremental manner, triggered by smaller units of utterance than sentences, so that even incomplete utterances can be interpreted. We also show that control management in a spontaneous dialogue can be fruitfully analyzed by regarding a speech act as comprising two parts – domain description and action description – and that control passing related to chargedness of a constituent.

1. INTRODUCTION

In dialogue, the fundamental aspects of the intentions of the participants are conveyed by speech acts of the utterances in the dialogue. Speakers are engaged in various types of speech acts other than just those frequently encountered. The speaker often tries to make sure that his interlocutor has received the information that he is conveying, that his intentions are understood, and that his attitudes toward his interlocutor and the general surrounding environment are clear. In a spontaneously spoken dialogue, these various speech acts are not necessarily expressed in full-fledged well-formed sentences; they are often couched in rather fragmentary, truncated semi-sentences or phrases.

Consider, for example, the following piece of dialogue collected from our corpus:

(1) E1: Reservations, Eric speaking.
T1: ah, yes, Eric.
E2: yes
T2: uuhh, I want to, I need to, uh, book a room
E3: OK
T3: in your, your hotel
E4: OK
T4: let me tell you what I need
E5: OK.

This excerpt shows many aspects of spontaneous dialogue that are relevant to the analysis of speech acts in real dialogue. One thing to be noticed is that the speaker often makes mistakes in generating phrases. For example, at T3, the speaker **repeats**, for some reason, the same word "your" twice, which is grammatically unnecessary. Or at T2, the speaker changes his mind and switches from "want" to "need", leaving the first verb phrase incomplete. This can be seen as a case of **repair**; the speaker thought "need" to be a more appropriate word than "want". Or, again at T2, the speaker uses the **filler** "uh", breaking the verb phrase. Notable in this example, or in any natural dialogue, is the frequent occurrences of backchannels "OK" and "yes", which are often the cause of unconnected or incomplete sentences, as in T2 and T3 where the intended sentence is broken into two phrases by the backchannel "OK".

This example also shows an important feature of spontaneous spoken dialogue: the system of **turn-taking** (Sacks *et al.* 1977). Turn-taking is often explained in terms of a **turn-constructional unit (TCU)**, at whose conclusion occurs what is designated as a **transition-relevance place (TRP)**, a place where it is appropriate for a new speaker to take over. As can be seen from the example above, a turn-constructional unit is not necessarily realized as a sentence, but as various syntactic constructs, ranging from lexical, clausal, phrasal, to sentential. Thus an important feature of spoken dialogue implies that non-sentential constructs play a significant role in conveying information. If dialogue can be considered a sequence of possibly overlapping turns, then TCU's may be thought of as the building blocks of dialogue, for turns are ultimately constructed from TCU's. Thus the nature of TCU's is crucial to the understanding of dialogue phenomena. However, although we acknowledge the importance of TCU, the concept of 'turn', or rather the term itself, can sometimes be confusing, as when it is controversial whether a backchannel constitutes a turn or not, and therefore, we use 'turn' to refer to a physical event, the sequence of utterances of one speaker delimited at both ends by the other speaker's sequences of utterances, and use **control** instead to mean 'floor' (Whittaker & Stenton 1988).

In order to take into account the characteristics of spontaneous dialogue, we have to find answers to several questions as to how the characteristics of spontaneous dialogue are related to the conveyance of speech acts:

- How speech acts are involved in such dialogue activities as grounding, turn-taking, and dialogue control;
- How each TCU is related to speech acts of an utterance;
- How TCU's are organized among themselves.

In the following, we will consider some ideas that may lead to the answers to the above questions.

2. CHARACTERISTICS OF SPONTANEOUS DIALOGUE

The limitation of applying the speech act theory in an overly simple manner to spontaneous dialogue can already be seen in the inherent grammatical properties of English. For example, a seemingly simple sentence:

(2) This is the cat

may actually be part of a longer utterance involving many verb phrases:

(3) This is the cat that killed the rat that ate the malt that lay in the house that Jack built.

This shows that without proper prosody it is rather hard to know in advance how phrases are to be discerned and hence how propositions are to be captured.

A similar, but more serious, case can be found in Japanese. Unlike English, Japanese is an agglutinative language with predominantly verb final sentence patterns. It is to be expected, therefore, that Japanese sentence units are rather easy to recognize — just look for the main verb. Textbooks and written forms of Japanese indeed vindicate such an expectation. Thus a typical textbook Japanese sentence would look like the following:

Sensei-ga	*hon-o*	*kai-ta*
Teacher-SUBJ	book-OBJ	write-PAST

'The teacher wrote a book.'

Since most of those sentences appearing in a grammar book are of this form, in which verbal forms come at the end, Japanese is typically considered an SOV language. On the other hand, such features as dropping case and conjunctive particles, frequent use of end particles and interjections, and abnormal ordering of words have often been considered signs of 'agrammaticism'. These facts seem to corroborate the view that in Japanese, grammatical sentence units are well established and rather easy to recognize.

When one observes Japanese in actual use, however, one can see that such a view is too simplistic, because a spontaneously spoken Japanese dialogue is full of features which are totally beyond the grasp of grammar books: phrases ending with conjunctions, interjections with verbs attached, unfinished utterances and other disfluencies.

In order to observe the true state of the language, we have been collecting task-oriented dialogues. Although we are mainly concerned with spoken Japanese, we are also collecting English dialogues for comparison (Kawamori *et al.* 1998). The corpus is still growing.

The Japanese corpus comprises 312 dialogues, whose total recording time is about 18 hours (1081 minutes). The recorded conversations are transcribed and the size of the resulting text is about 1 megabyte, which roughly corresponds to 500,000 characters in Japanese orthography and contains approximately 29,000 turns. The transcription is made using a tool that automatically segments speech data into units separated by pauses. Hence a turn is defined prosodically rather than syntactically or semantically. The topics of the conversations are mainly task-oriented; the participants are required to achieve a certain goal either in the dialogue or as a result of it. Tasks are not set up so that as much rigidity and artificiality as possible can be avoided. They are often provided realistic settings and motivations, such as following a direction, searching for a nice restaurant, or obtaining information about amusement parks. Most of them can be characterized as expert-client dialogues.

We selected 92 task-oriented dialogues from our Japanese corpus and observed what category of expression ends each turn: we wanted to see how grammatical each utterance actually is. The result is shown in tables 1 and 2. Since we used a semi-automatic means to

POS	Frequency	Percentage
Interjectories	4711	59.64%
Nouns	1918	24.28%
Case Particles	379	4.8%
Verbs	265	3.35%
Conjunctions	195	2.46%
Adverbs	152	1.92%
Adjectives	93	1.17%
Others	185	2.34%
Total	7898	100%

Table 1: Words at Turn-Start

	Frequency	Percentage
Incomplete	4646	56.42%
Complete	1823	22.14%
Nominals	1031	12.52%
Others	723	8.91%
Total	8234	100%

Table 2: Words at Turn-End

tag the parts of speech — human subjects checked the results of automatic morphological analyses — the categorization of parts of speech is based on conventional grammar.

Table 1 shows that more than half of the turns are started with interjectories. It is interesting to note that although about a quarter of the turns start with legitimately grammatical category, (nouns, adjectives, adverbs, or conjunctions), about 10% of them are started definitely ungrammatically, with verbs or case particles. If we regard interjectories as being "outside the domain of the well-formed sentence itself" (Martin 1975), close to 70% of the turns in our sample corpus are ill-formed.

Examples of case particles that often start sentences are *to* ('and') and *ga* ('subject marker'), as can be seen in the following:

(4) *ga* *dokora* *hen ni* *arimasuka?*
SUBJ where about exists ?
'(it) is where?'

Although it may not seem so strange as a spoken utterance, it is as grammatical a Japanese sentence, in the textbook sense, as an English sentence with a noun phrase containing only the definite article 'the'.

Sometimes a copular verb *da*, which is supposed to come at the end of a sentence, starts a turn:

(5) *da* *to* *omoimasu*
COP COMP think
'so I think'

Notice that these do not represent cases that occur rarely but rather utterances that are encountered quite frequently in daily dialogue.

If the beginning of a turn is often ungrammatical, the end of a turn is just as problematic. Table 2 shows what expressions come at the end of a turn. A turn is considered 'incomplete' if it has at the end such expressions as interjectories, case particles, conjunctive particles, incomplete verb forms, and other dependent elements. A more complete description of what ends an incomplete turn is given below:

INCOMPLETE	POS	FREQ	%
	Interjection	2909	62.61
	Case Particle	936	20.15
	Conjunctive particle	510	10.98
	Topic Particle	181	3.90
	Hypothetical	36	0.77
	Adjunct Noun	33	0.71
	Conjunction	18	0.39
	Demonstrative	10	0.22
	Adverbial	6	0.13
	Manner Adverb	4	0.09
	Numeral Suffix	3	0.06
	Total	4646	100

This table shows that 62% of the incomplete turns are ended with interjections. This figure may include those turns whose sole content is an interjection. Although whether an interjection counts as a sentence or not may be still controversial, most grammar books seem to agree with Martin to regard interjections as being outside the realm of grammar. 35% of the incomplete turns are ended with grammatically 'illegitimate' particles, such as topic particles, like *wa* and *mo*, and case particles. The demonstratives may well also be regarded as fillers, for such demonstratives as *ano* ('that') and *sono* ('it') are often used as fillers and *are* ('that one') is often used as an interjection.

On the other hand, a turn is considered 'complete' if it is ended with such an expression as a verb, an adjective, or an end particle. The following table gives what constitutes a complete turn.

COMPLETE	POS	FREQ	%
	End Particle	752	41.25
	Verbal Suffix	725	39.77
	Verb	178	9.76
	Copula	128	7.02
	Adj	35	1.92
	Adjectival Suffix	5	0.27
	Total	1823	100

Notice that the most frequent expression that ends a turn is an end particle, and this may show just that an end particle, despite its name, is not a good indicator of a complete turn. When an end particle comes at the end of a turn, we consider the turn to be complete; but this is an expedience we have to resort to for the sake of concreteness and objectivity. In fact, there are many turns that are considered complete in this sense but should be regarded as 'non-propositional'. The end particle *ne* ('you know'), for example, can be attached to almost any grammatical element, even to a filler, as in the following:

(6) *eeto* *desu* *ne*
FILLER COP EndP
'Well'

Although some may claim this is a grammatical sentence, doing so would imply a very formal definition of sentences, devoid of much sense as far as its practicality goes. Rather many would consider this as just a filler, rather than a sentence, and this shows that even in those cases where the turns are considered complete, according to the strict morphological analysis that we adopt here, they may well be considered somehow incomplete intuitively.

Although nominal endings are often used in writing for stylistic effects, these are not elements with which to finish a grammatically complete sentence. Hence, strictly speaking, a turn with a nominal at the end is not considered 'complete'. If we include turns with nominals at the end in the category of incomplete turns, then we see, again, that more than 70% of the units, or turns, are ill-formed.

This result shows how precarious the status of sentence is in spoken Japanese discourse, as mentioned at the outset, since it is possible that in at least more than half of the time, a turn does not consist of a sentence. This suggests that the discourse structure of dialogue cannot be understood in terms of the traditional grammar based on the concept of sentence (Dohsaka & Shimazu 1996).

All of these characteristics of spontaneous dialogue also put the traditional treatment of speech acts in rather an unfavorable position because speech acts, or more properly illocutionary acts, are usually regarded as consisting of an illocutionary force and a proposition. They further show that one needs a concept that is fine-grained enough to account for fragmentary utterances in actual data. We propose that a proposition smaller than a full-fledged ordinary proposition is involved in dialogue.

3. REPRESENTING DIALOGUE ACTS

As we said above, TCU's can be taken as the building blocks of dialogue, as seen more from its physical side; looked from its intentional, or speech act, side, the comparable concept would be **dialogue act**. Just as it is well-nigh impossible to think of smaller unit of utterance relevant to dialogue smaller than TCU, so is it hardly plausible to think of an intentional act relevant to dialogue smaller than dialogue act, since any small act can be assigned an dialogue intention and made a dialogue act. In the context of intentional, or semantico-pragmatic, aspect of speech act, we can treat TCU and dialogue act interchangeably. In this section, we consider how to introduce a means to give finer-grained representations of dialogue acts as called for by naturally occurring, spontaneous dialogues. For that purpose, we employ a mechanism similar to Discourse Representation Theory as introduced in Kamp (1981).

Conventionally, speech acts, or more properly illocutionary acts, are regarded as consisting of an illocutionary force and a proposition. In this view, an illocutionary force is comparable to a modality attached to the proposition. If we represent the proposition by P and an illocutionary force by $F(.)$ then an illocutionary act can be represented by $F(P)$(Searle 1969).

For example, consider the following sentence:

(7) John is a student.

If the propositional content of this utterance is represented as:

(8) Student(John).

When this sentence is uttered with the illocutionary force ASSERT, the entire illocutionary act would be represented as:

(9) $\vdash$ Student(John).

Or, consider another sentence:

(10) Open the door.

Supplying the unnamed subject of this utterance, one can represent the propositional content of this utterance as:

(11) Open(you, the_door).

When this sentence is uttered with the illocutionary force COMMAND, the entire illocutionary act may be represented as:

(12) ! Open(you,the_door),

where ! represents the logical operator corresponding to the illocutionary force COMMAND.

When the propositional content of the utterance is relatively clear and straightforward, such an understanding of illocutionary forces and speech acts is unproblematic, and most of the work in this area has been based on examples taken from neatly structured, well-formed sentences. When the propositional content is unclear or when the sentence in question is not quite so neatly constructed, however, there would be much reconstructing to do to apply this orthodox view of illocutionary acts. Notice that even in the simple case of (10), we assumed that we can safely understand the unmentioned subject of the utterance. Rather than taking speech acts to be carried by completely finished statements representing complete propositions, we need to think of them as composed of smaller parts whose incremental interpretations contribute to the overall interpretation of the speech acts themselves.

As has been observed above, an ordinary Japanese dialogue seldom consists of well-formed sentences like (7) that corresponds to a proposition. For example, in the following dialogue[1]:

(13) A1: well, on my desk
B1: Yes
A2: there are four books, aren't there?
B2: Ah, yes
A3: those books
B3: yes
A4: can you bring them to me?
B4: OK

The utterances like 'on my desk' (*boku no tukue no ue ni*) and 'that one' (*sore o*) do not correspond to sentences but phrases or parts of sentences. And they are the main candidates for TCU's, as can be seen from this example. When TCU's are to be interpreted in terms speech acts, then such dialogue acts usually correspond to semantic units smaller than propositions, which are presumably the content carriers of illocutionary force in the traditional sense.

If a sentence corresponds to a proposition, then these units smaller than sentences would correspond to parts of a proposition. One can call these parts of a proposition **subatomic proposition**[2]. If a stretch of discourse, consisting of several fragmentary utterances, proposes something then the discourse as a whole may be viewed as expressing the same speech act of proposal that would be expressed using some single well-formed sentence. If this single well-formed sentence represents a proposition, then each of the fragmentary utterances that constitute the stretch of discourse would be representing a part of the proposition, a subatomic proposition. And to make the parallel complete, each of the subatomic propositions would, with a certain type of, not necessarily illocutionary, force, correspond to **a subatomic speech act**, which is part of the act of proposal.

The term subatomic proposition is inspired by Parsons's (1990) subatomic semantics, the framework of semantics that studies entities beneath the level of atomic sentences. Parson's theory in turn is close in spirit to Davidson's (1980) event logic. Kamp's (1981) DRT is not only suitable for representing discourse, because it simulates the incremental way of interpretation, but is also good for representing subatomic propositions.

The speaker A's utterances in (13) consist of four turns, each of which is, in the traditional grammatical sense, not quite well-formed. In our analysis, these turns are understood as representing subatomic propositions.

The first turn, which is repeated here as (14),

(14) A1: *eeto, boku no tukue no ue ni desune*
well my desk upon
'Well, on my desk'

can be interpreted as a speech event in which something, whose identity is unknown yet, on the desk is mentioned. This speech event is describing an event in which unspecified entity on the desk is mentioned. Let us represent this described proposition as:

$$\begin{array}{|l|}\hline e_1| \quad \{x\} \\ \hline on(x, my\_desk) \\ \hline\end{array}$$

This in effect says that the utterance in question refers to a subatomic proposition, which is an event type, in which some type of object on my desk is described.

The next turn, which is repeated here:

(15) A2: *hon* *aru deshou* *yon satu*
book there-is isn't-there ? four
'There are four books, aren't there?'

says that the object in question is in fact a book, and there are four of them. The representation would be as follows:

$$\begin{array}{|l|}\hline e_2| \quad \{y,z\} \\ \hline \mathit{book}(y) \\ \mathit{card}(z,4) \\ [x=y=z] \\ \mathsf{Action}(\mathit{be}) \\ \hline \end{array}$$

Notice, however, that the 'illocutionary force' of these subatomic propositions is not really that of assertion or claim, but rather that of confirmation, as can be seen from the phrase *deshou* 'isn't there?' The whole point of these turns is to ascertain that the hearer recognizes that there is indeed such an entity, and to construct the mutual understanding that the speaker is talking about that entity. For the sake of clarity, let us write this in the following manner:

$$\begin{array}{|l|}\hline \text{CONFIRM} \\ \hline \begin{array}{|l|}\hline e_1| \quad \{x\} \\ \hline \mathit{on}(x, \mathit{my\_desk}) \\ \hline \end{array} \\ \\ \begin{array}{|l|}\hline e_2| \quad \{y,z\} \\ \hline \mathit{book}(y) \\ \mathit{card}(z,4) \\ [x=y=z] \\ \mathsf{Action}(\mathit{be}) \\ \hline \end{array} \\ \hline \end{array}$$

In a similar manner, the second part of the dialogue, with the third and fourth turns, can also be represented using the DRT notation.

$$\begin{array}{|l|}\hline \begin{array}{|l|}\hline e_3| \quad \{\} \\ \hline \mathsf{Obj}(\mathbf{dem}) \\ \hline \end{array} \\ \\ \begin{array}{|l|}\hline e_4| \quad \{\} \\ \hline [x=\mathbf{dem}] \\ \mathsf{Action}(\mathit{bring}) \\ \hline \end{array} \\ \hline \end{array}$$

These representations will be straightforward except for the following. Obj is introduced by the Japanese case particle *wo* designating the accusative case. **dem** is the demonstrative corresponding to 'them' in English and *sore* in Japanese, which picks out the most salient discourse referent.

Notice that both of these two portions are 'governed' by the two verbs that are related to actions; each of the subatomic propositions constituting the turns contributes to and culminates in the verb at each end. These actions can be taken to represent a sort of unit that corresponds to a speech act: the first corresponding to confirming and the second to requesting.

All told, the utterance of the speaker A would constitute the following speech act:

CONFIRM	REQUEST
e_1 | $\{x\}$	e_3 | $\{\}$
$on(x, my\_desk)$	$\mathsf{Obj}(\mathbf{dem})$
e_2 | $\{y, z\}$	e_4 | $\{\}$
$book(y)$ $card(z, 4)$ $[x = y = z]$ $\mathsf{Action}(be)$	$[x = \mathbf{dem}]$ $\mathsf{Action}(bring)$

Here CONFIRM and REQUEST are the dialogue acts of the corresponding utterances, or TCU's. Action(.) denotes the main verb of the corresponding unit. It would not be too difficult to show that this representation would correspond essentially to the speech act expressed by the following sentence:

(16) Can you bring me the four books on my desk?

If we just represent the whole sequence of utterances in our dialogue by the representation for (16), we would not be able to show that the dialogue is composed of the sequence of dialogue acts:

$$\langle [\mathsf{Action}(be), \text{CONFIRM}], [\mathsf{Action}(bring), \text{REQUEST}] \rangle,$$

which would be significant, as has been noted at the outset, for the analysis of spontaneous dialogue.

We have so far assumed as if the DRS's from A's utterances were constructed without any interference or help from other than A's own. In other words, we have ignored coordinative elements in dialogue. The source of such coordination in dialogue can be observed in two important aspects of spontaneous dialogue: feedback and control.

Feedback utterances are those utterances that are used to enable the participants of a conversation to exchange information about the essential features of the conversation that are presupposed by human communication. Allwood *et al.* (1991) give the following four features as essential basic functions of communication: **contact** – whether to continue to the interaction; **perception** – whether the message is perceived; **understanding** – whether the message is understood; **attitudinal reactions** – whether the message is accepted.

The interjectory utterances in Japanese, which are called *aizuchi* in Japanese, are often used for feedback. In fact, their main function is to let the interlocutor know that the speaker has accepted, understood, or perceived her message. They are also often used simply to show one's interest or willingness to listen.

Kawamori *et al.* (1994) give analyses of the ways in which *hai* ('yes') and other similar interjectory utterances are used and their analysis suggests that these utterances behave very much like the feedback utterances as explained by Allwood *et al.* (1991). The most frequent use of the Japanese feedback utterances is to give information on contact and perception. In other words, they are used to express the dialogue acts of 'acknowledgment' and 'prompt'. In our current sample dialogue, for example, B's feedback utterances of the same word *hai* can be interpreted as expressing three different dialogue acts, but 'acknowledgment' is the most frequent:

A1:	Well, on my desk	
B1:	Yes	[Acknowledge]
A2:	There are four books, aren't there?	
B2:	Ah, yes	[Positive reply]
A3:	Those books	
B3:	yes	[Acknowledge]
A4:	Can you bring them to me?	
B4:	Yes	[Accept]

This frequent use of 'acknowledgment' in dialogue is clearly related to grounding as emphasized by Clark (1992) and Traum (1994). An utterance of *hai* generally shows that the previous message has been grounded, or has become part of the common beliefs (Kawamori *et al.* 1994). In order to take this aspect of dialogue into account, DRS construction algorithm, or the representation of discourse itself, would have to be modified so that what utterance is used to introduce a certain idea into the common ground while a certain other utterance represents the content of the idea itself. A tripartite structure like $\langle K_A, K_B, K_{AB}\rangle$, may be required, where K_A and K_B are the first speaker's and second speaker's representation, respectively, of information, while K_{AB} is the combined representation of (suppositionally) common belief. Our construction of DRS in fact shows only the part which would be represented by K_{AB}.

Notice that of the two dialog act units above, the first one is governed by an existence verb *aru*, 'be'. This does not seem to be a particular point of the dialogue in question. The observation of the utterances in task-oriented dialogues from our corpus, shows that many such dialogue act unit pairs contain existence predicates or predicates presupposing existence. Indeed, we find many such examples in our corpus as in the following[3]:

(17) K1: I had telephoned for that reason
S1: yes
K2: Do you **remember**?

(18) S1: Do you still **have** that file?
K1: 'yes, I have. Can you hold on a second?'

Notice that **remember** presupposes the existence of an event or entity in the past, whereas **have** also presupposes the existence of the object of possession.

In other words, we can discern two components in a speech act in a task-oriented dialogue: one, which we call **discourse domain description**, that describes or identifies the domain in which the act in question is intended to be applied, and the other, which we call **action description**, that describes or identifies the action in question which is intended to be accomplished. Put more concretely, a speech act can be reinterpreted as "do something about watchamacallit", and that something is described by the action description of the speech act and that watchamacallit is identified by its domain description.

Given a speech act $\sigma = \langle \text{domain}, \text{action} \rangle$, if both its domain and action are fully described and can be identified in the context, we say that the speech act is **saturated**. If, on the other hand, either of the two components is not fully described or cannot be identified in the context, the speech act is said to be **charged**. If the underspecified component, either domain or action, of a charged speech act becomes fully described and the speech act saturated, we say that the underspecification is **discharged**.

The relationship between speech act in the traditional sense and the domain and action descriptions may be thought of as that between two different levels of communication. While speech act is rather static and constrained by convention, the domain action pair is more dynamic and spontaneous. The latter is concerned with informational status, rather like popular notions 'new information' and 'old information'; the charged component waits for new information, so to speak, whereas saturated component often serves as old information. In this regard, the domain-action pair may also be considered to be closely related to the notion of grounding, for one may say grounded information to be 'old' whereas information to be grounded is 'new'. One can even think of the domain-action pair to be an 'encapsulated' form of grounding process.

4. ANALYSIS OF CONTROL IN SPONTANEOUS DIALOGUE

The charge/discharge perspective introduced above can be used to analyze some control phenomena in spontaneous dialogue. In order to do so, we can set up some 'rules' about control-passing that rely on the charge/discharge perspective. Here the speaker with the control is designated as *S*, while the one without the control is *W*.

1. If *W* starts a **substantial contribution** to the discourse before *S*'s speech act is saturated, then *W* is INTERRUPTING.
2. When *S*'s speech act is saturated *W* can take control and start a substantial contribution to the discourse; this is not an interruption.
3. If *W* makes dialogue coordination contribution the control is not changed.
4. If, as after her speech act is saturated, *S* fails to give a substantial contribution to the discourse, the control is given to *W*.

Here we can think of utterances with substantial contribution as those other than backchannels and fillers, while dialogue coordination contribution is the kind of contribution (Clark 1992) that backchannels and fillers make.

With such a set of 'rules' for control-passing, we can analyze a passage like the following[4].

The excerpt, taken from our corpus, is about route-directing. The instructor, A, is describing to the novice, B, the way to a certain place in Tokyo area. The participants are speaking through telephone. The dialogue is not planned, nor is there any role-playing involved. To have an incentive, the novice is required to actually go to the place in question after this conversation. In the example below the following notation is used. CharX^n shows that speaker X has control at that turn and the speech act needs to be discharged, while DisX^n shows that speaker X is discharging unsaturated speech act with the superscript n so the control can be given to his partner. Intr$X$ shows that the turn is speaker X's interruption. $(X \to Y)$ shows that the control is given from speaker X to speaker Y. Sat$X$ indicates that X has control at that turn, and that there is nothing to be discharged, since the speech act is saturated.

CharA^1	A1:	if you pass that guard post
	B1:	yes
DisA^1	A2:	there is a pond on the right, and please go straight in front of it
	B2:	go straight seeing the pond on the right
	A3:	yes, go straight
CharA^2	A4:	then as there is Building No. 1
IntrB;		
CharB^2	B4:	Building No1
(A→B)	A5:	yes
DisB^2	B5:	but, it is where?
SatA	A6:	well, if you just follow the road you will see it right in front
SatB	B6:	oh, I will just hit Building No.1?
SatA	A7:	yes, that's right
(B→A)	B7:	OK

At A1, A is stating a conditional phrase and B acknowledges with 'yes'. Since A has not finished the conditional, his utterance is **charged** and the control is with A though B utters 'yes', which is not considered 'significant' in the above sense. At A2, the consequent of the conditional statement is uttered and A's statement is saturated. B2 repeats the second part of A2, to implicitly confirm. A3 further repeats part of B2, also implicitly confirming B2. Notice that A2 is of the form $\langle \mathsf{Action}(be), [\mathsf{Action}(go), \text{REQUEST}]\rangle$, exemplifying a typical saturated speech act. The sequence (A4,B4) is similar to (A1,B1);A4 is a subordinate clause giving a reason for action, and is charged because it is waiting for its main clause. The difference is that B4 constitutes a substantial contribution, since it contains information other than just coordinative. Hence, B4 is taken as B's interruption and B can be interpreted as attempting to take control. If A5 had contained more information than just feedback, then the control would have stayed with A, but in fact, A only says 'yes' (*hai*), and now the control is passed to B. This will start a sort of a substratum control, one level deeper than the original control, because A4 is still charged. Since B4 only contains a domain description, a noun phrase, it is now charged. B's speech act is discharged at B5 when he makes clear that he is requesting information about the domain, 'the building No1'. Turns from A6 to A7 are rather orderly sequence of saturated speech acts, and there is no reason to suppose that control should be passed. At B7, though, B utters 'OK' – *hai* in the original Japanese – which is not a substantial contribution, and hence, according to the fourth rule above, indicates that B has passed control to A. Notice that A4 is still waiting to be discharged, and so it can be said that control has returned to the original level. Notice also that the propositional content of A4 can be interpreted as $\mathsf{Action}(be)$, showing that A is describing the domain.

It can be seen from the description above that control mechanism does not have a monolithic structure, but rather highly stratified (Rieser 1999); it can have many different levels of control. We can discern a very local control, in the sense that a statement is charged as only a subatomic proposition can be interpreted by its utterance, and also a very global one, such as who is the main speaker of the dialogue, who is giving the instruction, or, in short, who has the upper hand.

It is also suggested that charge is derived from some sort of 'incompleteness': when only a subordinate clause is uttered, the constituent is charged; when only the domain of the dialogue is described, the dialogue is charged. This aspect of charge and the mechanism of control might have some bearing on the notion of topic and focus, but the subject is beyond the present work.

5. CONCLUSION

We have described some properties of spontaneous dialogue, which are beyond the 'grammaticality' of traditional grammars, that are concerned with turn-taking, or control passing, and turn constructional unit. We show that dialogue acts, as smallest units of dialogue, corresponding to TCU's, can be given logical representation in terms of DRT. Although our examples are mainly from Japanese similar effects can naturally be obtained for other languages, including English, as well. We have also shown that by looking at speech acts as involving domain/action components and charge/discharge functions, some of the properties of control passing can be fruitfully analyzed. We have also shown that control is stratified and not just a uni-level phenomenon.

Acknowledgements

We would like to express our sincere gratitude to the people who gathered at Amstelogue 99, especially to Jens Allwood, Hannes Rieser, Hans Kamp, and Mieke Rats for their insightful and helpful comments.

REFERENCES

Allwood, J., Nivre, J., & Ahlsén, E. (1991), 'On the semantics and pragmatics of linguistic feedback', Technical report, Gothenburg Papers in Theoretical Linguistics, 64.

Asher, N. (1993), *Reference to Abstract Objects in Discourse*, Dordrecht, Kluwer Academic Publishers.

Clark, H. (1992), *Arenas of Language Use*, Chicago, University of Chicago.

Davidson, D. (1980), *Actions and Events*, Oxford, Oxford University Press.

Dohsaka, K. & Shimazu A. (1996), 'A computational model of incremental utterance production in task-oriented dialogues', in *Proceedings of the 16th international Conference on Computational Linguistics*.

Kamp, H. (1981), 'A theory of truth and semantic representation', in J. Groenendijk, T. Janssen, & M. Stokhof (eds), *Truth, interpretation and information*, 1–41. Dordrecht, Foris.

Kamp, H. (1999), 'Utterances as transitions between attitudinal states', *talk presented at Amstelogue 99*.

Kawamori, M., Kawabata, T., & Shimazu, A. (1998), 'Discourse markers in spontaneous dialogue, A corpus based study of Japanese and English', in Manfred Stede, Leo Wanner, & Eduard Hovy (eds), *the Proceedings of Coling/ACL 98 Workshop on Discourse Relations and Discourse Markers*, 93–99.

Kawamori, M., Shimazu, A., & Kogure, K. (1994), 'Roles of interjectory responses in spoken discourse', in *the Proceedings of the 3rd international Conference on Spoken Language Processing*, 955–958.

Martin, S. (1975), *A Reference Grammar of Japanese*, New Haven, Yale University Press.

Parsons, T. (1990), *Events in the Semantics of English, a Study in Subatomic Semantics*, Cambridge, Mass., MIT Press.

Poesio, M. & Traum, D. (1998), 'Towards an axiomatization of dialogue acts', in J. Hulstijn & A. Nijholt (eds), *Proceedings of the Twente Workshop on the Formal Semantics and Pragmatics of Dialogues (13th Twente Workshop on Language Technology)*, 207–222, Enschede.

Rieser, H. (1999), 'The structure of task-oriented dialogue and the introduction of new objects', in *Preproceedings of Amstelogue 99*.

Sacks, H., Schegloff, E., & Jefferson, G. (1977), 'A simplest systematics for the organization of turn-taking for conversation', *Language* **50**, 696–735.

Schiffrin, D. (1987), *Discourse Markers*, Cambridge, Cambridge University Press.

Searle, J. (1969), *Speech Acts*, Cambridge, Cambridge University Press.

Suzuki, H. & Tutiya, S. (1991), 'A strictly incremental approach to Japanese grammar', in Jon Barwise, Mark Gawron, Gordon Plotkin, & Syun Tutiya (eds), *Situation Theory and Its Applications*, volume 2, 517–532. Palo Alto, CSLI.

Traum, D. (1994), *A Computational Theory of Grounding in Natural Language Conversation*, University of Rochester dissertation.

van Kuppevelt, J. (1995), 'Main structure and side structure in discourse', *Linguistics* **33**, 809–833.

Watanabe, M. (1974), *Kokugo Bunpouron (Theory of Japanese Grammar)*, Tokyo, Kasama.

Whittaker, S. & Stenton, P. (1988), 'Cues and control in expert client dialogues', in *the Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, 123–130.

NOTES

[1]The original dialog is in Japanese. The text only shows the English translations. Here is the glossed version of the same dialog.

A1: *eeto, boku-no tukue no-ue-ni desune*
well my desk upon
'Well, on my desk'
B1: *hai*
yes
'Yes'
A2: *hon aru-deshou yon-satu*
book there is, isn't there ? four
'There are four books, aren't there?'
B2: *a hai*
oh yes
'Ah, yes'
A3: *sore-odesune*
that
'Those books'
B3: *hai*
yes
'uh-hun'
A4: *tyotto motte kitekuremasuka?*
a little bring can you please?
'Can you bring them to me?'
B4: *hai*
yes
'OK'

[2]The concept of subatomic proposition may be considered related to the traditional grammatical notion of *bunsetsu*, which may be translated as 'sub-sentential clause', of Japanese, though the latter is specific to Japanese while the former is universal. According to Shinkichi Hashimoto (Watanabe 1974) a *bunsetsu* is "the shortest unit in a sentence when the sentence as language in actual use is divided into as many parts as possible." Although Hashimoto does not mention anything similar to the concept of speech act, his understanding of *bunsetsu* is intuitively similar to ours. Note, however, *bunsetsu* is part of an actually uttered sentence, whereas subatomic proposition is what is conveyed by such a unit. Although sometimes *bunsetsu* is also regarded as a phonological unit, as Martin (1975) defines it as an accent phrase, one might understand, in a sense, a *bunsetsu* as a means in Japanese to represent a subatomic proposition.

[3] The following are the original dialogs:

S1: *Sono-tame-ni denwa-sitan-dakedo*
for that reason telephoned-did-and
'I telephoned for that reason,'
K1: *ee*
yes
'yes'
S2: **Oboete**-*masu?*
Remember-POLITE?
'Do you remember?'

S1: *Ano fairu-wo mada* **motte**-*imasu?*
That file still **have**-be?
'Do you still have that file?'
K1: *ee motte-imasu tyotto matte-kudasai*
yes have-be a-little wait-please
'yes, I have. Can you hold on a second?'

[4]The original Japanese is as follows:

CharA[1] A1: *de sono syueizyo no mae wo sugiteitadakuto*
and that-guard in-front pass if-you-do
'if you pass that guard post'
B1: *hai*
yes
'Yes'

DisA[1] A2: *migi ni ike ga arimasuga sono mae wo tyokusin sitekudasai*
on-right pond there-is-but in-front-of-it go-straight please
'there is a pond on the right, and please go straight in front of it'

B2: *migi ni ike wo minagara tyokusin*
right pond while-looking go-straight
'go straight seeing the pond on the right'
A3: *hai tyokusin*
yes go-straight
'yes, to go straight'

(B→A) B3: *hai*
yes
'OK'
CharA[2] A4: *sousimasuto itigoukan ga arimasunode*
and-then building-No1 as-there-is
'then as you will see Building No. 1'

IntrB;

CharB^2 B4: *itigoukan*
Building-No1
'Building No1'

(A→B) A5: *hai*
yes
'Right'

DisB^2 B5: *ga dokora hen ni arimasuka?*
SUBJ where about there is?
'Where is it?'

SatA A6: *eeto soudesune mitinari ni kite itadakuto, mou*
well let's-see along-the-road if-you-come soon
sugu me no mae ni arimasu
right-in-front-of-eyes there-is
'well, if you just follow the road you will see it right in front'

SatB B6: *a itigoukan ni butiatarun desune?*
Oh Building-No. 1 hit isn't-it-the-case?
'oh, I will just hit Building No.1?'

SatA A7: *hai soudesu*
yes right
' yes, that's right'

(B→A) B7: *hai*
yes
'OK'

Masahito KAWAMORI
Media Information Laboratory
NTT Laboratories
3-1 Morinosato-Wakamiya, Atsugi,
Kanagawa, 243-0198, JAPAN

Akira SHIMAZU
School of Information Science,
Japan Advanced Institute of Science and Technology,
1-1 Asahidai, Tatsunokuchi, Ishikawa, 923-12, JAPAN

Location identification dialogues

Peter Krause
IMS
University of Stuttgart

September 1, 1999

Abstract

An architecture for the analysis of location identification dialogues is developed. Their pragmatics can be understood using a question-based analysis of identification dialogues as cooperative language games. The two participants are the informant, who introduces a reference object into the dialogue, and the informant, who inquires into the identity of the reference object.

A semantics for question and the DRT formalism for propositional attitudes and epistemic modalities form the semantic basis for the proposal.

Semantic relations between questions in context account for the appropriateness of questions in the dialogue. During the dialogue, further identification questions may arise. The goal of the language game is defined by the possibility for the identifier to introduce an anchor for the reference object in question, thus resolving the initial identification question among the questions under discussion.

A small set of move types that frequently occur in identification dialogues is proposed and related to the semantics.

The analysis scheme is illustrated by annotating a natural localization dialogue.

Contents

1 Introduction

In task-oriented dialogues, the question where a certain object is arises quite frequently. Be it that the object is needed as a tool for carrying out a plan, be it that the goal consists just in finding the object, understanding localisation dialogues is an essential ingredient in understanding task-oriented dialogue. Put more precisely, localization dialogues are question-answer-exchanges with the goal of identifying the location of an object precisely enough for some given purpose. Realistic localization dialogues present a number of interesting problems for semantic and pragmatic analysis. The semantics of spatial expressions, the semantics of propositional attitudes and the pragmatics of dialogue all interact.

In this paper, we present a pragmatic analysis of the following French localization dialogue which builds on the semantic analysis of (Krause et al., 1999)[1]. The dialogue has also been analyzed in (Asher, 1998).
A has had a bike breakdown on his tour into the Pyrenean mountains. He is describing his location to B using his mobile phone in order to get help from B, who can come with his car and repair material as soon as he knows where A is.

1. A: Je suis tombé en panne [avec mon vélo]. Peux-tu m'aider ?
   A: I have got a breakdown [with my bike]. Can you help me ?
2. B: Où es-tu ?
   B: Where are you ?
3. A: Je suis devant le refuge qui se trouve à environ un km après Couiza.
   A: I am in front of the hut which lies roughly one km after Couiza.
4. B: Il y a plusieurs refuges aux alentours de Couiza. Dans quelle direction es-tu parti de Couiza ?
   B: There are several huts in the surroundings of Couiza. In which direction have you left Couiza ?
5. A: Je suis sorti par la route Paul Sabatier. Puis j'ai roulé vers la montagne. À une clairière j'ai tourné à droite.
   A: I have left town on the Paul Sabatier road. Then I drove towards the mountains. At a clearing I have turned right.
6. B: Au grand carrefour ?
   B: At the big intersection ?
7. A: Non, après, là où on commence à avoir une belle vue sur la mer.
   A: No, afterwards, where one starts to have a beautiful view of the sea.
8. B: Ah, je vois, au Rocher du diable.
   B: Oh, I see, at the Rocher du diable.
9. A: C'est possible, il y avait un gros rocher.
   A: That's possible, there was a big rock.
10. B: Donc tu es à la Maison de l'aigle. J'arrive tout de suite.
    B: So you are at the Maison de l'aigle. I'll come at once.

This dialogue contains many interpretation problems. The utterances contain cases of ellipsis and vague relations from spatial prepositions and movement verbs. The utterance *je vois* in turn 8 cannot be interpreted literally. However, a concrete DRT analysis of the logical forms of the contributions, together with a sketch of the syntax-semantics interface based on lexical entries in λ-DRT and an account of presupposition justification for some critical cases has been developped in (Krause et al., 1999).
In this paper, we focus on the pragmatics of the dialogue. Location identification dialogues are analyzed as identification games. The rules in identification games are defined with reference to semantic representations that constitute a

[1]For the convenience of the reader, an English translation has been added

scoreboard, and to semantic relations between questions and answers and between different questions. These relations can be seen as an approximation of the notion of dialogue structure.
An architecture for the analysis of location identifying dialogues is developped in section (2). Identification dialogues are described as language games in section (3). In section (4), the example dialogue is analysed. Section (5) concludes.

2 An architecture for localization dialogues

Localization dialogues unite features of discourse about space and movement with features of dialogue. Research on spatial discourse based on formal semantics (see (Asher et al., 1995) for an overview) has shown a number of modules that are needed for this task.
Besides a compositional syntax-semantics interface and basic discourse semantics, there are otherimportant components. Detailed lexical information for spatio-temperal expressions, in particular motion verbs and spatial and temporal adverbs, is needed as a starting point for the construction of semantic representations. Because presuppositions are very frequently encoded in these lexical entries, a theory of presupposition is needed. In order to justify these presuppositions, world knowledge and knowledge about spatial concepts is needed in realistic cases.
Suppose these modules were already in place[2]. The specific problems of *dialogue analysis* would then still be open: what is the function of a given utterance in the dialogue ? and what is the connection between the semantic analysis of an utterance, the semantic representation of the context and its function in the dialogue ?
This paper proposes to add a pragmatic level of analysis on top of a semantic analysis as just described in such a way that the interactions between semantics and pragmatics become clear. The first part of this dialogue oriented analysis is the representation of the propositional attitudes of the speaker and the hearer. The second part consists in the semantic representation of questions in the context of propositional attitude representations. Together, they form the components of the scoreboard of a language game of cooperatively identifying a location that one of the participants has in mind.
Here, we concentrate on the aspects of localization dialogues that are not specific to spatial discourse. We view localization dialogues as instances of identification dialogues. This is only possible if the particular linguistic and spatial problems are relegated to other modules.
On the basis of an axiomatization of space-time, a syntax-semantics interface, and a resolver for anaphora and presupposition, the relevant spatial information

[2] In (Krause et al., 1999), such an analysis is worked out for the example dialogue from the introduction.

can be extracted from movement descriptions. All information about temporal relations is abstracted over. The result can be analyzed as an *identification dialogue* about a spatial referent one of the participants has in mind.
Identification dialogues are characterized as games in the next section, using an interpretation for identification questions and the DRT formalism for propositional attitudes (Kamp, 1990), (Asher, 1986).

3 Identification dialogues

There are different sorts of identification dialogues. Sometimes the identifier already knows the range of possible identifications (as in certain riddle solution dialogues), sometimes this is not the case (as in some clarification subdialogues). An identification dialogue is centered around the problem of identifying a reference. An *identifier* inquires about a reference made by the *informant.* Dialogues arising during the solution of riddles and clarification subdialogues in conversation are examples of identification dialogues. Riddle solution dialogues are not fully cooperative, clarification subdialogues are. Localization dialogues are special clarification (sub)dialogues. They are both cooperative and goal-oriented. The goal-orientedness has the consequence that all questions in the dialogue help to rule out alternative solutions for the identification problem. All answers are intended to contribute to the identification. The cooperativity implies that the participants have to coordinate their actions. In particular, the identifier has to give feedback whether the utterances of the informant contribute to the identification task. These feedback moves often are epistemic statements about the identifier's own information state, and include the report of eventual identification.
Identification dialogues end when the identification task has either been solved or an inconsistency arises. This has to be signalled by the identifier.
Before we can state the rules in detail, some assumptions about localization dialogues must be made explicit.
In the scenario of the example dialogue (1) the identifier and the informant aren't at the same location. The use of deictic expressions is limited to the first and second person pronouns[3]. To simplify the situation slightly, let's assume that the identifier has all the relevant spatial information from a very detailed map and that he has represented this information propositionally in a discourse representation structure. All the discourse referents in this DRS are anchored to objects in the map. An identification of a discourse object talked about by the informant in terms of one of these entities is therefore definitive. The question *But which object is this ?* makes no more sense after such an identification. This ensures

[3]These references will be resolved immediately in the example analysis at the end of the paper.

that the identification game terminates[4].

3.1 The hierarchical structure of questions and subquestions in identification dialogues

Identification dialogues start with an utterance by the informant implying the existence of the referent in question (the feeder)[5]. The initial identification question has the role of a topic-constituting question. Further questions have to be useful with respect to the topic-constituting question. Identification dialogues thus instantiate the architecture for dialogue analysis proposed in (van Kuppevelt, 1991) and (van Kuppevelt, 1995). Their restricted setting facilitates formalization.

3.2 Identification dialogues as games

Identification dialogues can be described as a language game, roughly in the sense of (Lewis, 1979). There are two participants, the informant, A, and the identifier, B. The scoreboard consists of the mental states of A and B[6]. Most of the moves in the game are linguistic ones, but there are also the non-linguistic moves that occur when the participants silently draw inferences. Their possible actions are specified by describing their preconditions in terms of the scoreboard, and their effects in terms of changes to the scoreboard.
The mental states are represented using the representational apparatus of DRT. Relative anchors (sometimes called internal links) are used to record identification assumptions that the identifier makes. There are two basic types of move: assertions and questions. All of the assertions of the informant, except confirmation moves and an indication that he does not have any (more) information about a question, are answers to questions of the identifier. Answers have to be informative with respect to a question the identifier has asked, but the informant is free to provide more information about other open questions after addressing the question just asked. The assertions of the identifier indicate that an identification cannot yet be made, indicate his understanding when he makes an identification, report an identification by redescribing the identified object. All of the questions, except the initial topic-constituting question, have to contribute to resolve that question (they have to be *useful*). Assertions are characterized using a DRS K, questions are characterized using a logical form for identification questions, Q.
The following idealizations have been made: the participants have fixed roles, e.g. only the identifier can ask questions. This is like in the interrogation games

[4]The question how to establish termination conditions for dialogues without this idealization is beyond the scope of this paper. See (Boër and Lycan, 1986) for discussion.

[5]The discourse referent does not need be introduced by a definite noun phrase. A specific indefinite is another possibility.

[6]For simplicity, in examples we will restrict our attention to the evolution of the identifiers' attitudes.

defined by (Groenendijk, 1999). We assume that the identifier is omniscient about the domain and that both participants only have and transmit true information.

3.3 The scoreboard

Basically, the scoreboard consists in the common ground and the private information states of the two participants, which contain relative anchors for the discourse referents in the common ground[7]. Because of the fact that each participant only has part of the information, we will speak of an identifier's and an informant's position. In the case of the identifier, the common ground can be seen as a knowledge ascription to the informant. An identifiers position in the language game of lazy identification dialogues is a DRS K_{id}, the knowledge of the identifier B containing a knowledge ascription to the informant A with content K_A, a list of relative anchors linking discourse referents from K_A to discourse referents declared in K_B and a pair of a topic-constituting question and a list of open subquestions. A subset of the discourse referents declared in K_B is singled out as *anchored*. The content of the anchors is not specified in the examples shown below, because for the identification task, it does not matter which information the identifier takes to be part of the anchor.
See section (4) for an example for the first 3 components[8].
An informant's position consists in his own knowledge represented by a DRS K_{inf} at the current stage of conversation[9], and a representation of the raised questions like for the identifier.

3.4 A precise account of the scoreboard

The scoreboard consists of the identifier's and the informant's part. The scoreboard S_B of the identifier B is a quadruple

$$\langle K_1, CG, A, K_2 \rangle$$

Here K_1 is a DRS representing the initial knowledge of B. All discourse referents in K_1 are assumed to be anchored. The internal anchors aren't specified, however, because it is a difficult problem to decide how much information should be part

[7]Properly speaking, two copies of the common ground would have to be distinguished. The rules of the language game ensure that the identifier's and the informant's version of the common ground never diverge. It is therefore unproblematic to use just one copy with identical variables in the representation of the examples. We also ignore this detail in the definitions.

[8]The DRS describing the beliefs of the informant will include, at any stage in the conversation, also the information contributed by the identifier up to that point. It actually plays the role of a common ground.

[9]In the simplified model of truthful conversation adopted here, it is not necessary to distinguish the contributions of the identifier. The informant will take over all assertions of the identifier, and accept all presuppositions that he makes.

of the anchoring conditions[10]. CG is a DRS representing the common ground. It contains discourse referents for questions that have been introduced during the conversation[11] and the content of assertions and presuppositions made by the informant and the identifier during the conversation. A is a list of anchors linking discourse referents in CG to internally anchored referents in K_1. K_2 is a DRS representing extensions to B's knowledge during conversation, for instance information about questions that is not part of the common ground, and discourse individuals that are inferred from information in the common ground. The parts of the identifier's scoreboard will be referred to using $S_B : K_1$, $S_B : CG$, $S_B : A$, $S_B : K_2$, and mutatis mutandis for the informant's scoreboard.
Let A' be the DRS with empty universe that contains an equation for each anchor in A[12]. Let us denote the combined DRS $K_1 \circ A' \circ CG$ by K^{com} for further reference[13].
The scoreboard of the informant has the same structure. It will be assumed throughout that the versions of the common ground of the two participants differ maximally in that one may be an alphabetic variant of the other. The rules specifying the effects of game moves ensure that the CGs cannot diverge. This is of course an idealization[14].

3.5 Representation language

The components of the scoreboard are DRSs. The standard basic DRT syntax for DRSs K is used in both the linear notation and the two-dimensional notation. In addition, the following constructs are used:

- α K (presuppositions)
- $(WHx\ K_1)K_2$ (WH-questions)
  WH-questions only occur in the following form:
  $(WHx\ K_1)\alpha\ [y|P(y)]; x = y$ (identification questions)
- $K?$ (decision questions)
- $Q; Q = qf$ (question referent introduction)

[10]In the application to localization dialogues, one option is to let the internal anchors specify coordinates on a map. This guarantees unique identification of spatial entities.

[11]In this sense, it functions like a stack of questions under discussion

[12]For instance, if A contains $RELANCH(x_{A1}, y_2), RELANCH(x_{A2}, y_5)$, then A' is

$$\boxed{\begin{array}{l} \\ x_{A1} = y_2 \\ x_{A2} = y_5 \end{array}}$$

[13]This combined DRS is the basis for the definition of some semantic notions below. $\circ$ denotes DRS merge

[14]In particular, it has to be assumed that the participants always accept each other's moves.

- $RELANCH(x, y)$ (relative anchors)
- $\diamond K$ (epistemic might)

Above, qf is the representation of a decision question or a WH-question. Identification questions of the form above will be abbreviated by $IdQ(y)$, the identification question about y, when the precise way the question is formulated does not matter[15].

3.6 Semantics

First, the standard update semantics for the basic DRT language with a presupposition operator and the epistemic modality is repeated (Def. 1). A similar definition can be found in (Dekker, 1993), except for the presupposition clause. The might-operator $\diamond$ has been introduced by Veltman (see (Veltman, 1996)). Then the semantic notions for questions are defined.
In the following, σ denotes a set of pairs of worlds and partial variable assignments, or an information state. The minimal state σ_0 is $W \times \{\epsilon\}$, where ϵ is the empty assignment[16].

Definition 1 (Update semantics for DRT with might)

- $\sigma[\![x_i]\!] = \{\langle w, g\rangle \mid \exists g' \langle w, g'\rangle \in \sigma \textit{ and } g' \prec_{x_i} g \textit{ and } g(x_i) \in \mathcal{U}\}$
- $\sigma[\![R(x_1 \ldots x_n)]\!] = \{\langle w, g\rangle \in \sigma \mid \langle g(x_1) \ldots g(x_n)\rangle \in F(R)(w)\}$
- $\sigma[\![x_1 = x_2]\!] = \{\langle w, g\rangle \in \sigma \mid g(x_1) = g(x_2)\}$
- $\sigma[\![K_1; K_2]\!] = \sigma[\![K_1]\!][\![K_2]\!]$
- $\sigma[\![\neg D]\!] = \sigma - \sigma[\![D]\!]$
- $\sigma[\![K_1 \Rightarrow K_2]\!] = \{i \in \sigma \mid \forall i_1 \in \{i\}[\![K_1]\!] \; \exists i_2 \; i_2 \in i_1[\![K_2]\!]\}$
- $\sigma[\![\diamond K]\!] = \begin{cases} \sigma \textit{ if } \sigma[\![K]\!] \neq \emptyset \\ \emptyset \textit{ otherwise} \end{cases}$
- $\sigma[\![\alpha(D)]\!] = \sigma, \textit{ if } \sigma[\![D]\!] = \sigma, \textit{ and undefined otherwise}$

Now for question interpretation. There are decision questions and identification questions. Both kinds of questions give rise to alternative values. Decision questions can be answered by either yes or no (the third possibility of answering *I don't know* will not be considered important for ascribing a semantic value). The

[15]E.g. for the identifier under certain conditions
[16]See (?)

semantic value of a decision question will be considered a pair of the semantic values of the positive and of the negative answer. This value can be calculated with respect to the logical form of the question alone.
The relevant semantic value for identification questions is the set of possible anchors. This makes the semantics of identification questions depend on the information state in which the discourse entity in question is declared and on the information state in which an anchor is sought. Because the informant in general does not have access to the alternative identifications that the identifier considers, the semantic value for identification questions is only available to the identifier. The semantic value for identification questions can only be calculated with respect to the complete scoreboard of the identifier.

Definition 2 (Question semantics)

- $alt(K?) = \{[\![K]\!], [\![\neg K]\!]\}$
- $alt(IdQ(x_{Ai}), S_B) = \{v \in Dom(K_1) | \, \sigma_0[\![K^{com}; \Diamond v = x_{Ai}]\!] \neq \emptyset\}$

The semantic value of a question term is identified with the value of the alt function:
$[\![qf]\!]^S = alt(qf, S)$

The clause for WH-questions specifies that any candidate that might be the correct anchor for the variable in question belongs to the alternatives.
For decision questions, the informant is on the safe side if he answers by K or its negation. For identification questions, the best he can do is to provide information *about* the referent in question. Only the identifier will be able to determine when the question is resolved.
The resolvedness of a question for the identifier plays an important role for the termination of the game. The following constructs are used: $resolved(Q_j)$ (the question Q_j is resolved), $KNOW(IdQ(x_A))$ (the identifier knows who x_A is). These notions depend on the entire scoreboard S_B of the identifier.
The definition of resolvedness applies both to decision questions and to identification questions:

Definition 3 (Resolvedness)
A question Q is resolved with respect to the scoreboard S iff $alt(Q)$ is a singleton set.

The definition of *knowing which* is a special case.

Definition 4 (Knowing which)
The identifier knows the answer to the identification question $IdQ(x_A)$, i.e. he knows who x_A is, written $KNOW(IdQ(x_A))$, iff the question is resolved.

The notion of usefulness is the most important tool for analysing the structure of the dialogue in terms of questions as in Van Kuppevelt 1991. A question is useful for another question if an answer to the first question potentially narrows down the alternatives for the second question.

Definition 5 (Usefulness of a question to an identification question) *Let Q_2 be a yet unresolved identification question.*
useful(Q_1,Q_2) wrt S_B if

- *Q_1 an identification question $IdQ(x_A)$ and*

  $$\exists v \in alt(Q_1, S_B) : alt(Q_2, S_B \circ RELANCH(x_A, v)) \subset alt(Q_2, S_B)$$

- *Q_1 is a decision question K? and either*

  $$alt(Q_2, \langle K_1, CG, A, K_2 \circ K \rangle) \subset alt(Q_2, \langle K_1, CG, A, K_2 \rangle)$$

  or

  $$alt(Q_2, \langle K_1, CG, A, K_2 \circ \neg K \rangle) \subset alt(Q_2, \langle K_1, CG, A, K_2 \rangle)$$

The definition of usefulness requires that at least one successful answer to the useful question truly narrows down the alternatives for the superordinate question. This notion of usefulness is quite weak: it does not imply that for any answer to the subordinate question, alternatives for the superordinate one are ruled out. But, given that the identifier can't know which answer is given, and given that an identification question with many useful but just one useless alternative will usually be more efficient to ask than a yes-no question based on a guess that will definitely narrow down the range of alternatives, this weak notion of usefulness seems the best option.

3.7 Summary of the Game Moves

The moves of the dialogue game are described in terms of preconditions and effects. The preconditions are mainly semantic conditions of quality. When they are fulfilled, then semantically the move is supported by the scoreboard. The effects are rules that determine how the two participants should change their scoreboard after the move.
First, a brief summary of the rules, then, in subsection (3.8) detailed definitions.

The identifier B He puts the topic-constituting identification question Q_{id} to the informant, $\text{ASK}_{tcq}(Q_{id})$. Whenever an identification question is resolved, he must introduce a relative anchor in his representation (IDENTIFY(RELANCH(x_A,x))) and signal this to A, possibly using an expression like *I understand* or *aha* (UNDERSTAND) and necessarily using a redescription (REPORTID) of the identified entity to let A know which entity was identified[17]. Identification moves may not be made if the identification question is not yet completely resolved. After such a non-linguistic IDENTIFY-move has been made, the identification question must be removed from the scoreboard.[18]. If, on the other hand, an identification question is not yet resolved after an answer of A, B may either simply continue by asking a question that is *useful* with respect to it (ASK_{subq}(Q)), or first signal that he does not yet know the answer, saying, e.g. *I don't know (yet) what* x_A *is* or *There are several P.*. This move is denoted by NOTYETID(p). A then has to take up the new question, which might again constitute a (sub-)topic, if it was an identification question. If no anchor for x_A remains *possible*, B must complain using UNID(x_A)[19]. However, on the idealizations assumed here this situation never can arise.

The informant A He initiates the dialogue by an assertion INIT(p) that functions as a feeder ((van Kuppevelt, 1995)) by setting up a discourse referent that can form the object of an identification question. When asked a question Q, he must provide a proper answer ANSWER(Q,p). Having done this, he may go on adding assertions that must be proper answers to some open identification question Q_{id}. After a REPORTID move of B, A may CONFIRM it, asserting consistency of the information contained in the identification with his own information, and then provide feedback (FBID) about which of this information he shares.

3.8 Preconditions and effects

The preconditions of the different moves are defined semantically. Assertions must be supported by the information state of the speaker.
The moves are denoted by expressions with logical forms as arguments.

3.8.1 Asking a topic-constituting question

[17]This is necessary, otherwise A would not be in a position to assess whether his assertions still contribute to the identification task.

[18]This corresponds to Van Kuppevelt's Dynamic Principle of Topic Termination or to Ginzburg's QUD-downdating

[19]Again, the argument must be specified by a definite description in this construct

This move is denoted by ASKtcq. The topic-constituting question is always an identification question. It is of course put to the informant by the identifier.

Definition 6 (ASKtcq)

Expression examples:

Where are you ?
Which red block ?

Syntax:

$$ASKtcq([WhichxP(x)]x \;=\; \alpha[y \mid Q(y)])$$

Preconditions:

- $\alpha : [y \mid P(y); Q(y)]$ *has a unique antecedent with respect to the common ground* CG.
- *The only antecedent, z, is not yet anchored (no condition of the form* $RELANCH(z,x)$ *is part of the scoreboard component* A_B*)*
- *No identification question about z is already part of* $B : K_2$.
- *The question may not yet be resolved.*

Effects:

- *The domain of the common ground is extended with a new question variable* Q.
- *The condition* $Q \;=\; [WhichxP(x)]x \;=\; \alpha[y \mid Q(y)]$ *is added to the common ground.*
- *The condition tcq(Q) is added to the common ground.*

The last effect serves to distinguish the topic-constituting question from other identification questions that may be entered later.
Expression examples:

Where are you ? (from turn 2 of the example dialogue)
Which red block ? (clarification question in a multi-modal interface)

3.8.2 Asking a subquestion

This move is denoted by ASKsubq. This move can appear as an identification question or a yes-no-question. It is played by the identifier.

Definition 7 (ASKsubq)

Syntax:

1. *ASKsubq(*$[WhichxP(x)]x = \alpha[y : Q(y)]$*).*
2. *ASKsubq(*$K?$*)*

Preconditions:

- *The subquestion is useful for an (unresolved) question in the common ground.*

  Formalization: $useful(Q, \alpha : [Q \mid unresolved(Q)])$
- *If it is an identification question, then the uniqueness presupposition is satisfied.*

Effects:

- *The domain of the common ground is extended with a new question variable* Q*.*
- *The condition* $Q = [WhichxP(x)]x = \alpha[y \mid Q(y)]$ *is added to the common ground.*

3.8.3 Signalling that an identification question is not yet resolved

This move is denoted NOTYETID. It may be played by the identifier.

Definition 8 (NOTYETID)

Syntax:

- *NOTYETID(*$\neg KNOW(B, IdQ(\alpha : [y|P(y)]))$*)*
- *NOTYETID(*$\exists x_1[\exists x_2[NEQUx_1x_2 \wedge \Diamond x_1 = y \wedge \Diamond x_2 = y]]$*)*

Preconditions:

- *The identification question Q about the y answering to $\alpha : [y|P(y)]$ is in the common ground.*
- *Q is not yet resolved*

Effects:

The informant is obliged to answer the identification (again) in the next turn.

Expression examples:

There are several huts near Couiza. (turn 4)
I don't know which hut you mean. (natural alternative to turn 4)

3.8.4 Identifying a referent

This move is denoted by IDENTIFY. It is non-linguistic. The effect is the introduction of a relative anchor in the representation by the identifier.

Definition 9 (IDENTIFY)

Syntax:

IDENTIFY(x_A,x_B)

Preconditions:

- *x_B is the only element in the domain of $S_B : K_1$ such that $K^{com} \circ x_A = x_B$ is consistent.*
- *The identification question about x_A in the QUD is resolved, if it was present.*

Effects:

The anchor $RELANCH(x_A, x_B)$ is introduced into $S_B : A$.

3.8.5 Signalling understanding

This move is denoted by UNDERSTAND.

Definition 10 (UNDERSTAND)

Syntax:

UNDERSTAND($\alpha[Q|idqu(Q)]$)

The argument is usually left implicit. The condition idqu/1 singles out identification question as opposed to decision questions.

Preconditions:

An identification move (IDENTIFY) has just been made on good grounds by the identifier.

Effects:

The informant accommodates in his version of the common ground that an identification question has been resolved. If there is only one identification question under discussion, then he can resolve the anaphoric condition.

Expression examples:

I see.
I understand.
Aha.

3.8.6 Reporting an identification

This move is denoted by REPORTID and can only be played by the identifier.

Definition 11 (REPORTID)

Syntax:

REPORTID($\alpha[x_A|P(x_A)]$,$\alpha[x_B|Q(x_B)]$)

Preconditions:

- *An identification has been made, which may or may not already have been signalled to the informant using UNDERSTAND.*
- *The identification is of x_A as x_B.*
- *The condition $Q(x_B)$ determines a unique antecedent with respect to $S_B : K_1$.*
- *The condition $\lambda x Q(x)$ is not entailed by what is said about x_A in the common ground.*

Effects:

- *The informant marks the identification question about x_A as resolved.*
- *The informant is now obliged to confirm the identification if possible.*

Expression examples:

So you are at the Maison de l'Aigle. from turn 10.
You mean the red pyramid on top of the green block.

3.8.7 Reporting complete identification failure

This move is denoted by UNID. It can be played by the identifier.

Definition 12 (UNID)

Syntax:

UNID($\alpha : [x_A | P(x_A)]$)

The parameter is usually left implicit.

Preconditions:

$S_B : K^{com}$ *has become inconsistent.*

Effects:

The identification game is over with the negative result that no identification is possible.

3.8.8 Initializing the conversation

This move is denoted by INIT. It consists in any assertion with content K setting up a discourse referent about which an identification question can be asked. There are no relevant preconditions. The effect is simply the extension of the common ground by K.

3.8.9 Answering a question

There are several cases - answering a yes-no-question and answering an identification question.

Definition 13 (ANSWER)

Syntax:

- *ANSWER(αQ,YES), ANSWER(αQ,NO).*
- *ANSWER(αQ,K), where K is any DRS about the referent in question.*

A DRS is about a referent if it presupposes it.

Preconditions:

- *The question which is answered is present in the common ground.*
- *For answers to decision questions: the answer entails the positive or the negative alternative.*
- *For answers to identification questions: the answer is about the referent to be identified. This means that the answer contains a definite description that must be resolved to the same referent as the presupposition of the identification question with respect to the common ground. The answer must provide information about the anchor of this object in the informant's information $S_A : K_1$.*

Effects:

- *Extension of the common ground with the new information.*
- *The identifier must check whether the question is now resolved.*

3.8.10 Confirming a reported identification

This move is denoted by CONFIRMID.

Definition 14 (CONFIRMID)

Syntax:

CONFIRMID($\alpha[x_A|P(x_A)]$, $\alpha[x_B|Q(x_B)]$)

Preconditions:

- *The identification of x_B as x_A has just been reported.*
- *The addition of the information presupposed by the report to the CG version of the informant does not cause an inconsistency in $S_B : K^{com}$*

Effects:

The identification assumption can be strengthened from defeasible to undefeasible, but this is not explicitly modelled in the current account of the language game.

An expression example is in turn 9:
That's possible.

3.8.11 Providing feedback about an identification report

This move is denoted by FBID.

Definition 15 (FBID)

Syntax:

FBID(K)

Preconditions:

- *The identifier has reported an identification. The identification has just been confirmed.*
- *The DRS K is an existential statement describing an object with properties compatible with those mentioned in the identification report, that are not entailed by those.*

Effects:

The information contained in K is entered into the common ground.

Turn 9 provides an expression example:
There was a big rock.

3.9 Starting a game

The game is started by an INIT move, any assertion made by the informant that introduces a discourse entity into the common ground.

3.10 The termination conditions

An identification game is over if the initial, topic-constituting identification question is resolved for the identifier and if his identification has been confirmed by the informant. Note that in the final state of the game, there may remain pending identification questions that have been raised as subquestions of the topic-constituting identification question. This is an important difference between topic-constituting questions and subquestions.

4 Dialogue representation

After the description of the dialogue game, we can come back to the analysis of a concrete example. The contributions have been annotated by the move types. In most cases, it is not difficult to recover the arguments of the dialogue moves. This information has not been made explicit for space reasons.

4.1 The annotated example dialogue

The move types described above can be used to annotate the example dialogue from the introduction.

1. INIT
   A: Je suis tombé en panne [avec mon vélo]. Peux-tu m'aider ?

2. ASK$_{tcq}$
   B: Où es-tu ?

3. ANSWER
   A: Je suis devant le refuge qui se trouve à environ un km après Couiza.

4. NOTYETID
   B: Il y a plusieurs refuges aux alentours de Couiza. Dans quelle direction es-tu parti de Couiza ?

5. ANSWER
   A: Je suis sorti par la route Paul Sabatier. Puis j'ai roulé vers la montagne. À une clairière j'ai tourné à droite.

6. ASK$_{subq}$
   B: Au grand carrefour ?

7. ANSWER
   A: Non, après, là où on commence à avoir une belle vue sur la mer.

8. UNDERSTAND, REPORTID
   B: Ah, je vois, au Rocher du diable.

9. CONFIRMID, FBID
   A: C'est possible, il y avait un gros rocher.

10. REPORTID
    B: Donc tu es à la Maison de l'aigle. J'arrive tout de suite.

The second sentence of turn 1 and the last sentence of turn 10 are not part of the identification dialogue proper and have therefore not been annotated.

4.2 Example representation

The following DRS serves to illustrate the result configuration of the local identification dialogue of section (1). In this case, no identification questions are left, so the question stack is not shown[20]. The representation shows the identifications made and the belief attributed by B to A about the spatial configuration at the end of the dialogue[21]. Temporal information and information about eventualities has been left out (the spatial information must be inferred from a full representation containing eventualities).
The components $S_B : K_1$, $S_B : A$ and $S_B : CG$ have been drawn together in the obvious way.

[20]The full history of the example dialogue fills more than 15 pages.

[21]The following abbreviations are used: *dist* is a function returning the distance between two locations, *end* yields the final location of a directed path, *start* is analoguous, $prefix$ is the relation of being an initial path, the concatenation symbol $\circ$ is used to concatenate paths, *changedir* relates to consecutive paths with different directions, *gc* is an abbreviation for *grand carrefour*, *cbv* abbreviates the complex state of affairs expressed by *commencer à avoir une belle vue*

$$(K_{B0})$$

$$\begin{array}{c}
\text{RELANCH}(l_{A1},\, l_1)\ \text{RELANCH}(p_{A1},\, p_1)\\
\text{RELANCH}(d_{A1},\, d_2)\ \text{RELANCH}(l_{A2}, l_2)\\
\text{RELANCH}(l_{A4}, l_3)\ \text{RELANCH}(l_{A3}, l_5)\\
\text{RELANCH}(p_{A5}, p_3)\ \text{RELANCH}(l_{A3}, l_5)\\
\text{RELANCH}(p_{A6}, p_5)\ \text{RELANCH}(p_{A3}, p_6)\\
\text{RELANCH}(p_A, p_{A1} \circ p_{A2} \circ p_{A3})\\
\text{RELANCH}(l_A, l_6)\ \text{RELANCH}(r_A, r_3)
\end{array}$$

$$\text{ATT}(\text{CG}, \left\{ \left\langle \text{KNOW}, \boxed{\begin{array}{c}
r_A,\, l_A,\, l_{A1},\, l_{A2},\, l_{A3}\ l_{A4},\, p_A,\, p_{A1},\, p_{A2},\, p_{A3},\, p_{A4},\, p_{A5},\, p_{A6},\, d_{A1}\\
\hline
\text{Couiza}(l_{A1})\\
dist(r_A, l_{A1}) = 1km\\
\text{loc}(\text{A}, r_A)\\
\text{on}(p_A, l_1)\\
end(p_A) = r_A\\
prefix(p_{A0} \circ p_{A1} \circ p_{A2} \circ p_{A3}, p_A)\\
\text{rps}(p_{A1})\\
dir(p_{A1}) = d_{A1}\\
l_{A2} = end(p_{A1})\\
start(p_{A2}) = l_{A2}\\
\text{changedir}(p_{A1}, p_{A2})\\
\text{versmontagne}(p_{A2})\\
end(p_{A2}) = l_{A3}\\
\text{clairiere}(l_{A3})\\
start(p_{A3}) = l_{A3}\\
adroite(p_{A2}, p_{A3})\\
\text{continuation}(p_{A4}, p_{A2})\\
start(p_{A4}) = l_{A3}\\
\text{gc}(l_{A4})\\
l_{A3} \neq l_{A4}\\
\neg\ \text{gc}(l_{A3})\\
p_{A2} = p_{A5} \circ p_{A6}\\
start(p_{A5}) = l_{A2}\ end(p_{A5}) = l_{A4}\\
start(p_{A6}) = l_{A4}\ end(p_{A6}) = l_{A3}\\
cbv(l_{A3})
\end{array}} \right\rangle \right\})$$

The anchors have been introduced when the corresponding identification question was resolved. The extended construction history of the representation shows how each subquestion helps to solve B's identification task.

5 Conclusion

Localization dialogues were analyzed using identification dialogue games. The rules of these games were specified using definitions based on a standard semantic formalism. This provides a bridge between the pragmatic level of language games and the semantic level. The analysis scheme is successful to the extent that naturally occurring example dialogues can be naturally analysed using it. The deeper question whether the rules of the game themselves can be predicted from more general assumptions about pragmatics has not been addressed.
Future work will concern mixed-initiative identification dialogues and the specific granularity problems that arise in spatial discours.

References

Asher, N. (1986). Belief in Discourse Representation Theory. *Journal of Philosophical Logic*, 15(2):127–189.

Asher, N. (1998). Varieties of discourse structure in dialogue. In Hulstijn, J. and Nijholt, A., editors, *Proceedings of Formal Semantics and Pragmatics of Dialogue*, pages 1–9. University of Twente.

Asher, N., Aurnague, M., Bras, M., Sablayrolles, P., and Vieu, L. (1995). De l'espace-temps dans l'analyse du discours. Technical report, Institut de Recherche en Informatique de Toulouse.

Boër, S. and Lycan, W. G. (1986). *Knowing Who*. MIT Press.

Dekker, P. (1993). *Transsentential Meditations*. PhD thesis, University of Amsterdam.

Groenendijk, J. (1999). The logic of interrogations. In *Proceedings of SALT*, Santa Cruz. UCSC.

Kamp, H. (1990). Prolegomena to a Structural Theory of Belief and Other Attitudes. In Anderson, C. and Owens, J., editors, *Propositional Attitudes – The Role of Content in Logic, Language, and Mind*, number 20 in Lecture Notes, pages 27–90. CSLI.

Krause, P., Reyle, U., and Schiehlen, M. (1999). Spatial inferences in a localization dialogue. In Bras, M. and Vieu, L., editors, *Proceedings of the Workshop* Sémantique et Pragmatique du Dialogue, Corsica. ATALA.

Lewis, D. (1979). Score-keeping in a language game. In Bäuerle, R., Egli, U., and von Stechow, A., editors, *Semantics from different points of view*. Mouton-de Gruyter.

van Kuppevelt, J. (1991). *Topic en Comment.* PhD thesis, University of Nijmegen.

van Kuppevelt, J. (1995). Discourse structure, topicality and questioning. *Linguistics*, 31:109–147.

Veltman, F. (1996). Defaults in update semantics. *Journal of Philosophical Logic*, 25:221–261.

Modelling Questions and Assertions in Dialogue Using Obligations

Jörn Kreutel
SAIL LABS S.L.
Barcelona

Colin Matheson
Language Technology Group
University of Edinburgh

Abstract

The model of questions and assertions proposed here assumes that the notion of DISCOURSE OBLIGATIONS provides a useful way of looking at some aspects of the way these acts drive parts of dialogue. Our approach to obligations largely follows that proposed by Traum and Allen ([8]). We argue that obligations may provide a structuring mechanism for dialogues which is able to account for at least some of the data handled by Ginzburg's QUESTION UNDER DISCUSSION (QUD) (Ginzburg 1997 [3]).

1 Introduction

This paper investigates the use of DISCOURSE OBLIGATIONS in modelling questions and assertions in cooperative dialogues, largely following the work of Traum and Allen [8]. We accept that the notion of obligation is required in order to explain some facts about the behaviour of dialogue participants (DPs), and we suggest that an extended use of obligations can account for some aspects of 'cooperativity'. Section 2.2 below looks briefly at the background to the use of discourse obligations.

The main focus of attention concerns the issues raised by Ginzburg concerning the analysis of questions (see Ginzburg 1997 [3]) in which the notion of QUESTION UNDER DISCUSSION (QUD) is introduced. We suggest that, given that obligations are independently required, a slightly more extensive use can cover at least some of the facts which motivate the QUD, and hence that the latter may not be necessary as a separate device.

As far as coverage is concerned, the model is closely related to the theoretical approach described in Poesio and Traum (1997) [6], and we therefore assume that we cover the range of data discussed by Poesio and Traum. It should be noted that, in order to focus on the issue of discourse obligations, we do not explicitly address the phenomenon of

GROUNDING here (see, for example, Traum 1994 [9]); however, this simplifying assumption is not necessary, and the mechanisms proposed by Traum are entirely compatible with our model.

In general, we are not attempting to provide a complete theory of assertions and questions in dialogue, but we do claim to model some essential aspects of the way such acts contribute to the way the dialogue is driven forward without extending the range of descriptive means employed by Poesio and Traum. Most of the main aspects of our analysis have been implemented and applied to a significant range of simple dialogues in order to confirm the practicality of the model (see Kreutel 1998 [4] for details of the implementation and for a proposal for dealing with more complex phenomena like indirect answers or clarification subdialogues). Although the approach described here may not therefore be novel in the sense that it captures new phenomena, we believe that the analysis does represent a streamlining of certain aspects of dialogue modelling, that it is practical, and that it sheds new light on some established issues. In particular, the general framework established by Traum and Allen and by Poesio and Traum does not yet offer an extended analysis of questions, and thus our analysis represents a significant advance in this particular field of dialogue modelling.

2 Information States

One interesting development in recent dialogue modelling is the use of INFORMATION STATES (ISs) to characterise the state of each participant's knowledge as the discourse progresses. We therefore accept here the suggestion that some aspects of dialogue modelling are best captured by providing a detailed description of the 'state the dialogue is in' along with a full account of the possible update mechanisms (which can be seen as dialogue MOVES and/or ACTS, depending on the context of the discussion). We adopt a notion of IS that follows some of the proposals by Poesio and Traum, who provide an extensive representation of ISs which incorporates a model of 'grounding' (Traum 1994 [9]), provides a reinterpretation of speech acts and illocutionary force (Austin 1962 [1]; Searle 1969 [7]), and contains a formal representation of obligations. (Traum 1994 [9]; Poesio and Traum 1997 [6])

For present purposes, we assume that an IS simply consists of:

1. A representation of the DIALOGUE HISTORY, DH, as a sequence of moves.
2. A representation of the obligations imposed on the DPs, where we assume a separate obligation stack for each DP and a stack of collective obligations.

The following sections detail our assumptions about the contents of ISs, beginning with the kind of actions the DPs may perform.

2.1 Dialogue Acts

We adopt Poesio and Traum's idea that moves in a dialogue involve the performance of actions at multiple levels of discourse structure, and that these actions can be decomposed into DIALOGUE ACTS. Following Poesio and Traum (1998) [5], we capture the idea that any speech act may appear in a wider discourse context and thus subclassify dialogue acts into CORE SPEECH ACTS and ARGUMENTATION ACTS, where the latter characterise the context-dependent actions of core speech acts. Here, we assume the following kinds of dialogue acts:

Core Speech Acts

- 'Forward-Looking' Acts: `assert` and `ask`, which correspond to the performance of assertions and questions.
- 'Backward-Looking' Acts: `accept`, `reject`, `accept_answer`, and `reject_answer`, which refer to the actions of accepting or rejecting an assertion with respect to its 'assertive' and its 'answerhood' properties. In contrast to the `reject_answerhood` argumentation act defined below, we assume that a `reject_answer` act is performed by an asker in situations where an askee has expressed an inability or unwillingness to provide an answer, and this statement is accepted by the asker.

Argumentation Acts

- `answer`, `reject_init`, and `correct`, which indicate that an `assert` act is meant to provide an answer, to express the askee's inability to come up with an answer, and to correct a previous assertion.
- `info_request`, which is performed if an askee performs an `ask` act as a reply to a question.
- `reject_answerhood`, which an asker performs by means of an `assert` or an `ask` when an `assert` which is meant to provide an answer does not resolve the asker's question.

The following section outlines the motivation for the use of obligations in dialogue modelling and the particular assumptions we make about the way the above dialogue acts may introduce or retract obligations.

2.2 Obligations *vs.* Intentions

A comprehensive introduction to the notion of obligations in dialogue modelling can be found in Traum and Allen (1994) [8]. We adapt the latter proposals to cover some aspects of discourse which are typically analysed using intentions, accepting the argument that modelling every aspect of dialogues using intentions is too restrictive. The main problem with this approach is, briefly, that intentions are typically analysed as persistent and achievable goals (Cohen and Levesque 1990 [2]), but in question-answer contexts, in which it is assumed that answering results from the recognition and adoption of the asker's intention, this analysis fails to account for cases where a DP cannot or will not provide an answer; for instance:

(1) A[1]: Did Pete drive here?
B[2]: I don't know.

Assuming A's intention is to get B to provide an answer, B cannot adopt this as an intention because it is unachievable. Similar problems arise when B chooses not to provide an answer. In such cases there is no intention-based explanation for why an askee responds at all. Traum and Allen thus argue for a notion of obligation in discourse; cooperative DPs are socially *obliged* to respond, no matter what their intentions are.

However, as Traum and Allen assume that obligations are immediately discarded once an attempt to satisfy them has been made, they need to refer to intentions in cooperative question-answer sequences if an answer does not immediately satisfy the asker's desire for information.[1] We, on the contrary, assume that the obligation introduced by an `ask` act (the obligation to `answer` the question that has been asked) is only discarded when the asker has accepted an answer (`accept_answer`) or when the askee expresses an inability to provide an answer, in which case the asker will perform a `reject_answer` act. As we will show below, this assumption enables us to model the actions of cooperative DPs in significantly complex scenarios without referring to their intentions.

For `assert` acts, on the other hand, we assume that any reply will discard the obligation to `address` the assertion and thus preserve the analysis of assertions in an obligation-based framework proposed by Traum and Allen, and subsequently by Poesio and Traum. We further assume that a `correct` argumentation act introduces a collective obligation on the DPs to `resolve` the conflict in their beliefs.

Given this view of the way obligations are introduced and retracted in the course of a dialogue we argue that it is possible to model most of the data covered by Ginzburg's QUD using obligations.

3 Modelling Questions and Assertions

An important aspect of Ginzburg's analysis of questions and assertions in dialogue (for example, Ginzburg 1997 [3]) is the use of a partially ordered set of representations of questions, called the QUESTION UNDER DISCUSSION. Whereas in Poesio and Traum's model it is the obligations which determine the possible responses a given move in a dialogue will elicit, Ginzburg proposes a version of IS in which this central role is assigned to the QUD.

In essence, the QUD is a device for structuring particular aspects of the representations of dialogues; Ginzburg suggests that elements of the QUD represent the issues which characterise segments of discourse and which, together with the common ground[2] which comprises a representation of the history of the discourse, determine a set of admissible follow-up moves at any state of a dialogue. A question in the QUD thus represents a 'live issue' (Ginzburg 1997 [3]) which is not resolved until certain conditions are true. The kind of data Ginzburg adduces are exemplified in the following discourse. This shows a case where a discussion arises, during which the question of 'who will come' is assumed to persist until A's acceptance of B[2] in A[5]:

(2)
	A[1]:	Who will agree to come?
	B[2]:	Helen and Jelle.
	A[3]:	I doubt Helen will want to come after last time.
	B[4]:	Nah, I think she's forgiven and forgotten.
	A[5]:	Ok.

Our concern here is to show how the data which the QUD is designed to handle may be covered by an extended use of obligations, which we assume are independently motivated as discussed above. The following subsections therefore look briefly at the formal representation of the QUD and at the issues of how elements in the QUD arise, how they are resolved, and how they are subsequently downdated. After an outline of the updating principles we assume for ISs we can then show in general how our model accounts for Ginzburg's case.

3.1 The Question Under Discussion

Ginzburg assumes that there are two main ways in which a question arises and may become 'under discussion':[3]

- An assertion of p raises the question *whether* p
- A question q introduces q

It is not *necessary* in Ginzburg's framework for assertions and questions to add questions to the QUD, of course; the contents of an assertion may be accepted, in which case no discussion takes place, and questions may be rejected as topics.

As for the removal of elements from the QUD, Ginzburg uses a fairly simple principle:

- A question q that is an element of the QUD will be removed from the QUD if the common ground of the DPs contains information that *resolves* q relative to the mental state of that DP whose acting has given rise to q.[4]

For an element of the QUD introduced by a question, the resolution can be provided by the propositional content of an assertion performed as the answer to the question. On the other hand, an element in the QUD which results from a DP's not immediately accepting an assertion *p1*, might, for instance, be resolved by an assertion *p2* which provides evidence for *p1*.

As far as the admissible actions following a change of turn are concerned, Ginzburg assumes that for replies to non-conjoined assertions and questions the turn-holder will address the first element in the QUD.[5].

We accept the general argument that without some kind of structuring mechanism as provided by the QUD it is difficult, if not impossible, to characterise the way in which topics are managed in the course of a discourse. However, we suggest that obligations can provide the necessary structure, and the following subsection looks briefly at the kind of updating mechanism we employ.

3.2 Updating Information States

We currently assume three update 'scenarios' which cover replying to an assertion, responding to a question, and evaluating the answer to a question. We leave aside here issues concerning the representation of knowledge and belief, which determine the particular response which will be performed, and restrict ourselves to specifying the conditions which characterise the respective scenarios and to listing the admissible follow up moves which update ISs.

Responding to an assertion

Conditions:

1. DP1's first obligation is to `address Move`
2. `Move`, an assertion by DP2, is the first element in the DH

Depending on his or her actual beliefs, DP2 will `accept` the assertion, reply with an `ask` act which requests evidence, or – in the case where there is a conflict of beliefs – perform an `assert` act which `corrects` the initial assertion. A request for evidence will, in addition to imposing an obligation to `answer`, introduce a conditional to the DH: If DP1 accepts a later assertion as the required evidence, the original assertion will also be `accept`-ed. If DP2 cannot come up with an answer, the assertion will be `reject`-ed.[6] An assertion which `corrects` the assertion, on the other hand, will introduce a collective obligation on the DPs to resolve the conflict between their respective beliefs. The following discourse fragment provides an example:

(3)
A[1]: Helen was at the party yesterday.
B[2]: How do you know?
A[3]: Her car was there.
B[4]: Ok.

Here, B's move in B[2] adds a conditional to the DH expressing the fact that B will accept A[1] if A can come up with an answer for B[2]. The *Ok* in B[4], in this context, expresses B's acceptance of A[3] with respect to its assertive properties *and* as an answer for B[2], i.e. as evidence for Helen's being at the party. The antecedent of the conditional is then present in the DH, which triggers the acceptance of A[1].

Replying to a question

Conditions:

1. The DP's first obligation is to `answer Move`

Here, DP will either perform an `assert` act, which provides an `answer` or expresses an inability to answer (`reject_init`), or an `ask` act which functions as an `info_request`. The following discourse illustrates the second possibility:

(4)
A[1]: Was Alex at the party yesterday?
B[2]: Was her car there?
A[3]: Yes.
B[4]: Ok. She was there, then.
A[5]: Ok.

Here, our obligation-based approach provides an appropriate structuring mechanism that can account for the DP's actions. Consider B's utterance in B[4]: After A's move in A[3] there are two obligations imposeed on B: to `address` the assertion that Alex's car had been where the party took place (A[3]), and to `answer` the question whether Alex had been at the party (A[1]). Following our assumption that DPs will always try to satisfy the topmost element on their obligation stacks (stated in the specification of the scenarios above), B's *ok* both accepts A's assertion in A[3] and accepts it as an answer to his own question in B[2]. The obligation to address A[3] thus being dropped, the obligation to answer A[1] becomes topmost. As B's knowing whether Alex had been at the party was dependent on his knowing whether her car had been there, B can now answer A[1] and this part of the dialogue terminates with A's accepting the answer provided by B in B[4].

Evaluating the answer to a question

Conditions:

1. DP1's first obligation is to `address Move2`
2. DP2's first obligation is to `answer Move1`
3. The first element in the DH is the assertion of `Move2` by DP2, which either performs an `answer` or a `reject_init` argumentation act.

The latter conditions describe a situation in which the context of an assertion is checked to see if the assertion appears to be an attempt to answer a previous question. Here, we assume that DP1 first evaluates the assertive properties of `Move2` according to the scenario for evaluating assertions. If the assertion is `accept`-ed, DP1 then checks whether it resolves the question in `Move1`. If it does, an `accept_answer` act is performed and the obligation on DP2 is retracted. Otherwise, DP2 will either perform a `reject_answer` act, have DP1 come up with an alternative answer, or perform a further `ask` act whose resolution will conditionally trigger the resolution of `Move1`. In the latter two cases, we assume that a `reject_answerhood` act is performed simultaneously and that the obligation on DP2 to `answer Move1` persists. The dialogues below represent the two situations mentioned above in which a reply to a question is not accepted with respect to its answerhood properties (5) and its assertive content (6):

(5)

A[1]:	Helen wasn't at the party.
B[2]:	How do you know that?
A[3]:	Her car wasn't there.
B[4]:	But she could have come by bicycle.
A[5]:	I left at 5 in the morning and she hadn't shown up.
B[6]:	Ok.

Here, the sequence of moves B[2]-A[5] illustrates the effects of the `reject_answerhood` argumentation act: even though B accepts A's assertion that Helen's car had not been at the party, B does not consider this to provide evidence for Helen's absence. B's assertion in B[4], which performs the `reject_answerhood` act, means that the obligation introduced by his request for evidence (B[2]) is still imposed on A. As B[4] is an assertion, A has the option of starting a discussion subdialogue about its propositional content. However, the content of B[4] is accepted by A, who then tries to come up with an alternative answer for B[2]. This is because, after the obligation to address B[4] is discarded by A's implicit acceptance of the assertion, the obligation to answer B[2] becomes topmost again in A's obligation stack.

In contrast to the latter case, where an answer to a question is not accepted because it does not resolve the question, in the following example an assertion which is intended to answer a request for evidence is questioned with respect to its assertive properties. This situation results in two subdialogues dealing with providing evidence for an assertion being embedded in each other:

(6) A[1]: Helen wasn't at the party.
B[2]: How do you know that?
A[3]: Alex wasn't there.
B[4]: Are you sure?
A[5]: Yes. I didn't see his car.
B[6]: Ok. But Helen might nevertheless have been there.
A[7]: Fred wasn't there, either.
B[8]: Ok. Then you might be right.

3.3 Obligations and the QUD

The issue here is to show in general how the obligation-driven model covers the same ground as the QUD.[7] For questions, the presence of an element in the QUD corresponds exactly to the obligation to answer. We update and downdate the obligation representations in exactly the same way as the QUD is updated and downdated, namely when an `ask` act has been performed and when the askee comes up with an assertion which resolves the question, in which case the asker will perform an `accept_answer` act in our model.

If one compares our analysis of a dialogue like (4) with the one provided by Ginzburg, this 1:1 correspondence between obligations and elements of the QUD becomes obvious (see Ginzburg 1997:15 [3] and Kreutel 1998 [4]). The main distinction is that we currently assume that obligations are represented as stacks; the QUD as a partially ordered set is formally more flexible.

For assertions the correspondence is not so straightforward. We assume that the obligation introduced by an assertion of p is discarded by any kind of response. In Ginzburg's model, in contrast, the 'question associated with p' will be present in the QUD as long as the content of the assertion is under discussion. However, the notion of an assertion being under discussion is modelled in our framework by means of the conditional introduced by a request for evidence[8] and by the idea that replying to p with *not(p)* introduces a collective obligation on the DPs to resolve the conflict between their beliefs.

This way we are able to reconstruct the intuition that in discussions the content of an assertion continues to be a 'pending issue' even after the obligation introduced by the assertion has been discarded. For instance, if one considers the dialogue (6) above, the conditional introduced by B's request for evidence in B[2] will be present in the DH until B's *Ok* in B[8]. The persistence of the conditional captures the fact that the question whether or not Helen was at the party stays 'under discussion' in the course of the subdialogue initiated by B[4], which discusses the issue of whether Alex had been there or not. As for the removal of an element from the QUD, on the other hand, the resolution of a conditional in our model and the dropping of the collective obligation correspond to the question associated with p being dropped in Ginzburg's approach.

Finally, there is the question of how we reconstruct the DP's decision on which issue to address. Here, our assumption is that the topmost element in the turn-holder's obligation stack determines the relevant action, reflecting Ginzburg's proposal that the topmost element in the QUD will be addressed first (at least for non-conjoined questions and assertions).

4 Discussion

As shown in the preceding section, an obligation-driven approach to dialogue is in principle able to reconstruct most of the achievements of Ginzburg's notion of QUD as far as its ability to come up with a descriptively adequate model of dialogue structure is concerned. Additionally, the use of discourse obligations may overcome a certain lack of explanatory power exhibited by the QUD: even if it does account in an intuitive way for the issue of *how* DPs act at a given state of a dialogue, it nevertheless does not explain *why* they act as they do, and why they act *at all.*

While Ginzburg's model presupposes that DPs act cooperatively if they act according to the QUD update and downdate scenarios as quoted above,[9] our approach highlights what 'cooperativity' actually means, namely the DPs' willingness to act according to the obligations imposed on them.

In conclusion, our model preserves the explanatory power of Traum and Allen's concept of modelling uncooperative actions using obligations. However, for cooperative dialogues our modified assumptions about the way the obligations introduced by a question are managed allows us to account for situations in which a question is not immediately resolved in cases similar to those handled by Ginzburg, namely with no reference to the DP's intentions. We do not deny that intentions, alongside belief and knowledge, contribute to a DP's actual responses, but we nevertheless claim that obligations provide more powerful descriptive and explanatory means for reconstructing the essentially rule-governed behaviour exhibited by cooperative DPs.

Notes

[1] As noted in Traum's thesis (Traum 1994 [9] p.103), there are actually two representations of obligations in the theory, and it is the elements in the discourse obligations stack which are removed immediately. However, there is also a general notion of obligation which allows for general reasoning and which can be used to reimpose obligations in circumstances where the discourse obligation is assumed not to have been satisfied. The precise nature of this operation with respect to questions is unclear to us, however, and we argue that our approach represents an attempt at the required precision.

[2]'Facts' in Ginzburg's terms

[3]Questions can also arise 'inferentially' and in clarifications, but we shall concentrate on the central issues here.

[4]For a formal definition of a question being *resolved* see Ginzburg (1997) [3]).

[5]'Addressing' in Ginzburg's model is defined in terms of the semantic relations of 'aboutness' and 'dependence' which may hold between assertions or questions and the respective follow-up moves (Ginzburg 1997 [3])

[6]For the use of conditionals in the our framework see Poesio and Traum's analysis of how the acceptance of an assertion affects the common ground of the DPs, i.e. the DH in the model assumed here ([5]).

[7]For some examples, see Kreutel 1998 [4]

[8]Which would be a 'discussion' in Ginzburg's model. Our motivation for assuming two kinds of discussion (requests for evidence and 'real' discussions) is that in the former case a DP's failure to establish the propositional content of an assertion as mutually believed knowledge will *not* enforce belief revision, whereas 'losing a discussion' will.

[9]See Ginzburg (1997), which explicitly introduces the notion of 'cooperative' assertion and querying

References

[1] John L Austin. *How To Do Things With Words.* Clarendon Press, Oxford, 1962.

[2] Philip R Cohen and Hector J Levesque. Persistence, intention and commitment. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*, chapter 3. MIT Press, Cambridge/MA, 1990.

[3] Jonathan Ginzburg. Querying and assertions in dialogue. May 1997.

[4] Joern Kreutel. An obligation-driven computational model for questions and assertions in dialogue. Master's thesis, University of Edinburgh, Edinburgh, 1998.

[5] Massimo Poesio and David Traum. Towards an axiomatisation of dialogue acts. In *Proceedings of the Twente Workshop on Language Technology*, pages 207–222, 1998.

[6] Massimo Poesio and David R Traum. Conversational actions and discourse situations. Research Paper 92, University of Edinburgh, September 1997.

[7] John Searle. *Speech Acts: An Essay in the Philosophy of Language.* Cambridge University Press, Cambridge, 1969.

[8] David Traum and James Allen. Discourse obligations in dialogue processing. In *Proceedings of the 32nd Annual meeting of the Association for Computational Linguistics*, pages 1–8, June 1994.

[9] David R. Traum. *A computational theory of grounding in natural language conversation.* PhD thesis, University of Rochester, New York, December 1994.

Amstelogue'99 Proceedings

Workshop on the Semantics and Pragmatics of Dialogue
Amsterdam University 7-9 May 1999

Part II

editors:

Jan van Kuppevelt
Noor van Leusen
Robert van Rooy
Henk Zeevat

sponsors:

DISC(Esprit)
Dutch Research Foundation (NWO)
Faculty of Humanities (University of Amsterdam)
Institute for Logic, Language and Information
Dutch Research School on Logic (OZSL)
TRINDI(Language Engineering)

List of Contents

Part I

Part II

13. Hannes Rieser
The Structure of Task-oriented Dialogue and the Introduction of New Objects
email:rieser@lili.uni-bielefeld.de

14. Robert van Rooy
Permission to Change
email:vanrooy@hum.uva.nl

15. David Traum
20 Questions on Dialogue Act Taxonomies
email:traum@cd.umd.edu

16. Claus Zinn
Understanding Mathematical Discourse
email:zinn@informatik.uni-erlangen.de

Cognitive States, Discourse Structure and the Content of Dialogue

Alex Lascarides
Division of Informatics,
University of Edinburgh
alex@cogsci.ed.ac.uk

Nicholas Asher
Department of Philosophy,
University of Texas at Austin
nasher@bertie.la.utexas.edu

Abstract

In this paper, we focus on two puzzles concerning anaphora in dialogue. On the basis of these puzzles we argue for three criteria for solving them: the representation of dialogue content must include rhetorical relations since they impose different constraints on the antecedents to anaphora; for discourse interpretation to be computable, it must be highly modular, for example the logic for discourse update and the logic for cognitive modelling must be separate; but these different modules must be allowed to interact in complex ways. We specify an account which meets these three criteria within Segmented Discourse Representation Theory (SDRT: Asher, 1993). We illustrate our approach by first using principles or rationality and cooperativity to derive principles for computing discourse content—in particular, the rhetorical connections between the propositions. We then exploit these cognitively-motivated but discourse-level axioms to account for the puzzles concerning anaphora.

1 Introduction

In this paper, we examine anaphora in dialogue. Anaphora resolution is extremely complex, so for the sake of clarity we will focus on just two puzzles, illustrating them with a 'minimal pair' of examples, in order to bring out the major features of the framework. The first puzzle is that constraints on anaphora are not only sensitive to contextual information, but also to the way it's presented: compare (1ab′c), where the anaphoric expression *his name* sounds odd, to (1abc) and (1ab′c′), which are acceptable.

(1) a. A: How can I get to sixth street?
b. B: There's someone Downtown that you could ask.
b′. B: You can ask someone Downtown.
c. A: What's his name?
c′. A: Oh yeah? And what's his name?

Second, communicative intentions affect anaphora. For example, *2pm* in (2e) can't be 2pm on Saturday, even though Saturday is referred to in the previous utterance, because this is incompatible with the underlying intention: given (2d), why would A ask if he can meet at 2pm on Saturday? Furthermore, something more than intentions are needed, since *2pm* can't be on Sunday either.

(2) a. A: How about meeting next weekend?
b. B: That sounds good.
c. How about Saturday afternoon?
d. A: I'm afraid I'm busy then.
e. ??How about 2pm?

We'll argue for three criteria for dialogue interpretation. First, one needs rhetorical relations such as *Elaboration* and *Contrast.* Second, the framework must incorporate several levels of representation and reasoning. In particular, discourse content (both semantic and pragmatic) must be represented at a different level from cognitive states; and the logic for computing discourse content must have only restricted access to that content and to cognitive states. And third, one must account for complex interactions between the discourse and cognitive levels.

Most current theories of dialogue analysis fail to fully meet these criteria, and their accounts of (1) and (2) are problematic. We'll show here how framework of discourse semantics known as SDRT meets our requirements. Asher (1993) and Lascarides and Asher(1993) show how SDRT augments DRT with rhetorical relations and is modular in the way required, and Asher (1999) has added a separate module for limited reasoning about cognitive states and Gricean implicatures. In this paper, we defend the view that separate but interacting theories of discourse structure and cognitive states are essential to the analysis of dialogue. We illustrate our approach by first using principles of rationality to derive principles for computing discourse content—in particular the rhetorical relations that hold, and then exploiting these cognitively-motivated but discourse-level axioms to account for the constraints on anaphora observed in (1) and (2).

2 Dialogue and Mental States

Many researchers have emphasised cognitive modelling in analysing dialogue (e.g., Grosz and Sidner 1986, 1990, Litman and Allen 1990, Hobbs *et al.* 1993, Sperber and Wilson 1986). In these theories, the compositional semantics of an utterance is incorporated as a new belief, and this leads to further new beliefs or changes to old ones. These updates are viewed as pragmatic effects. So interpretation is belief update and revision, and cognitive states and discourse content are represented in the same module.

Moore and Paris (1993) and Asher and Lascarides (1998a) argue against this approach. First, the step from utterances to attitudes can only be made by default (Sparck Jones, 1989); interpreters have no direct access to other speakers' cognitive states. Further, as the *full* semantics of the utterance goes into a belief, the language for characterizing cognitive states must be at least first order. A serious problem, then, is that this view has a tractable model neither for belief revision nor even for belief update; this is because both will involve tasks equivalent to consistency checks over extensions of first order default theories. On this view, dialogue interpretation goes beyond what's recursively enumerable.

But we believe that an adequate competence model of dialogue interpretation should be (at worst) decidable. This is because dialogue interpretation is a rational activity that agents can engage in effectively. Agents can systematically and reliably predict how other agents interpret dialogue; indeed, if they could not, then successful communication would not take place. This means that any competent user of the language should agree on the information that a speaker intended to convey in his utterance, proviso that they agree on the properties of the discourse context. A competence model can explain this agreement, if the competence model provides a way of deciding if a particular inference about dialogue meaning is valid. If the logic underlying the competence model is undecidable, however, then no such procedure exists, even if there are unlimited reasoning resources. That being the case, the competence model would not, even in principle, provide a way of deciding whether an agent will infer a particular fact about the meaning of the dialogue or not, regardless of whether the logic validates that fact. So the competence model would fail to model a very important aspect of dialogue interpretation: the communal agreement on what the dialogue means.

Equating discourse interpretation with cognitive effects also leads to empirical problems: one can't explain why small surface changes affect anaphora, e.g., as in (1). One can't distinguish the intentions of the utterances (1abc) and (1ab′c) on independent grounds, because one can't with confidence distinguish the domain level plan that underlies (1abc) vs (1ab′c), or the underlying discourse level plan of speech acts.

To see this, let's consider the domain level plans first. One might try to distinguish them by assuming that the domain plan underlying (1ab′c) involves going Downtown and choosing someone *randomly*, whereas the choice isn't random in the plan underlying (1abc). But this places a huge burden on the axiomatisation of such fine-grained plan operators. One would have to assume plan recognition axioms that translate very small differences in surface form into dramatic effects on the plan operators that are invoked in the domain plan. We believe such a model would be highly impractical. At any rate, even if it were feasible, the differences in the anaphoric possibilities that are generated by (1b) vs (1b′) has a simpler, *monotonic* explanation, which bypasses domain plans altogether. We discuss this in section 3.

Now consider the discourse level plan of speech acts. Traditionally, these are computed via inferences about the speaker's beliefs, on the basis of his utterance (e.g., Perrault 1990, Grosz and Sidner 1990). But this means that one can't claim with confidence that the speech acts underlying (1b) vs. (1b′) are distinct, because one can't claim with confidence that B's beliefs which lead him to utter (1b) vs (1b′) are distinct. For example, it's possible that in each case, his belief that someone Downtown knows the way is *de dicto* (i.e., he doesn't have a particular person in mind). The fact that the belief could be *de dicto*, even if he utters (1b′), also undermines the previous conjecture, that the domain level plan for (1ab′c) may involve asking a particular person Downtown, as opposed to a random person.

Overall, an independently motivated axiomatisation of plans and intentions is too coarse-grained to provide the rich discourse structures that are necessary for constraining anaphora. So they overgenerate possible interpretations and predict discourse coherence where there isn't any.

3 Dynamic Semantics

How can these shortcomings be avoided? Dynamic semantics can help. In dynamic semantics, computing discourse content isn't equated with cognitive effects. Rather, one defines *discourse update*, i.e., the way the current utterance updates a representation of the meaning of the discourse context (e.g. DPL (Groenendijk and Stokhof, 1991), DRT (Kamp and Reyle, 1993) and SDRT (Asher 1993, Lascarides and Asher 1993)). Unlike Hobbs *et al.* (1993), discourse content is represented at an 'intermediate' level between compositional semantics and the attitudes: it expresses more content than compositional semantics since discourse update may resolve underspecified semantic conditions arising from the grammar, but it does not in general record cognitive effects. Indeed, cognitive modelling isn't traditionally addressed within dynamic semantics at all.

Dynamic semantics provides the basis for analysing (1). The difference in scope between the quantifier *someone* and the modality *can* in (1b) vs (1b′) yields, monotonically, the differences between (1abc) and (1ab′c) (Kamp and Reyle, 1993).

Moreover, one difference between (1ab′c) vs. (1ab′c′) is that there's a *Contrast* between (1c′) and (1b′) which is lacking in (1b′c) (note that intuitively, inserting *but* in (1c′) doesn't affect its semantic contribution to the dialogue). This rhetorical difference leads to different constraints on anaphora in SDRT (Asher, 1993), which predict that *his* refers successfully in (1c′). To see this, let's examine SDRT's analysis of *Contrast*. Roughly put, two propositions that are connected with *Contrast* must have a partial isomorphism between their hierarchical DRS-structures,[1] and at least one pair of nodes in this mapping must have opposite polarities or 'contrasting themes'. The details for computing contrasting themes are given in Asher (1993). But this isn't our main concern here. Rather, we focus on how the 'isomorphic' mapping affects anaphoric possibilities. Again, roughly put, there's often a choice of partial isomorphisms, and the highest ranked mapping is the one that's closest to a total isomorphism. Furthermore, a discourse referent that is introduced in a sub-DRS D_1 of the first proposition in the *Contrast* relation can act as an antecedent to an anaphor in the sub-DRS D_2 of the second proposition that D_1 is mapped to. This means *Contrasts* allows elements in embedded DRSs to be antecedents to certain anaphora, since D_1 may be embedded, and this is contrary to the general constraint of accessibility in DRT (and indeed, the fact that embedded DRSs are generally inaccessible explains why (1ab′c) is odd, as the discourse referent introduced by *someone* is in a DRS that's embedded within the scope of the modality *can*).

So, how does *Contrast* effect the anaphora in (1ab′c′)? Well, the embedded DRS D_1, that has a discourse referent introduced by *someone*, is mapped to some part of the DRS structure of (1c′). Now, (1c′) contains a λ-abstract introduced by *what*, and therefore the discourse referent introduced by *his* is in a DRS D_2 that's embedded within the scope of this λ operator. Indeed, *Contrast* here induces a total isomorphism, with D_1 mapped to D_2, since these are both embedded one step down the top level DRS. So the *Contrast* makes *someone* accessible to *his*. As we mentioned, it's inaccessible in (1ab′c), because of the different rhetorical relation that holds between (1b′c).

Dialogue (1) demonstrates that the quantifiers, modalities and rhetorical relations constrain anaphora. Dynamic semantics can define such constraints. But we must extend it to account for dialogue. Grice (1975) and the above AI work demonstrate that the interplay between what people say and the goals that motivate them is vital. Dynamic semantics must model this, and dialogue (2) demonstrates why. On the one hand, if (2e)'s underlying goal—to meet next weekend but not on Saturday afternoon—plays no role in constraining (2e)'s content, then *2pm* could be interpreted as 2pm on Saturday. On the other hand, if *only* the goals matter, then *2pm* would refer to 2pm on Sunday. The fact that *2pm* fails to refer demonstrates that both linguistic and cognitive constraints are necessary.

We draw several conclusions. First, since rhetorical relations constrain anaphora, our theory should include them and their effects on content. Second, if interpreting dialogue is to be reliable and computable, then one must represent dialogue content at a separate level from cognitive states. Moreover, given that computing rhetorical connections involves reasoning with partial information (Lascarides and Asher 1993, Hobbs *et al.* 1993), the logic that models this can't have full access to the logic of dialogue content or the logic of cognitive states, as this would make discourse interpretation undecidable. And finally, an adequate account of anaphora requires a precise model of the interaction between the discourse and cognitive levels.

Most frameworks of dialogue analysis fail to meet our three criteria. The cognitive approaches mentioned in §2 aren't modular in the way required. Grosz and Sidner (1986) analyse discourse at several levels, but they don't distinguish *Contrasts* from other coordinating moves and they don't distinguish the constraints on anaphora imposed by (1b) vs. (1b′). Standard dynamic semantic theories fail in other respects. They ignore rhetorical relations and the influence of goals and intentions on dialogue content, and therefore fail to account for (2).

Our main point is that a satisfactory interpretation of dialogue requires both dynamic semantics and cognitive modelling in *separate*, but communicating modules. To this end, we have extended SDRT with a module that links what people say to what prompted them to say it (Asher 1999).[2] we'll use general principles of rationality in this theory to compute axioms for inferring discourse structure. We'll then demonstrate that the resulting theory models the interplay between linguistic and cognitive information in the interpretation of (2).

4 Cognitive Modelling in SDRT

SDRT has two parts. First, there's a formal language for representing discourse content (Asher, 1993). This is an extension of DRT: a discourse is represented as an SDRS, which is a recursive structure of labelled DRSs, with rhetorical relations between the labels. Second, there's a theory known as DICE, which is used to compute rhetorical relations in the SDRS of a discourse from the compositional semantics of its clauses (Lascarides and Asher, 1993). DICE consists of a modal propositional language augmented with $>$ ($A > B$ means *If A then normally B*). Rules in DICE are of the form in (3), where $\langle \tau, \alpha, \beta \rangle$ means β is to be attached to α with a rhetorical relation (α and β label SDRSs), where α is part of the discourse context τ, *some stuff* is information about α and β, and R is a rhetorical relation:

(3) $(\langle \tau, \alpha, \beta \rangle \wedge \text{ some stuff}) > R(\alpha, \beta)$

DICE doesn't have full access to the language of SDRSs. If it did, then discourse interpretation wouldn't be computable. Rather, the conditions in an SDRS K_α get translated in DICE into predicates of the propositional variable α, where α labels K_α. So DICE has access to the *form* of information content, but not to all its entailments.

We turn now to model interactions between dialogue content and cognitive states. We'll first correlate attitudes with utterances, and then derive DICE axioms from basic principles of rationality. Dialogue utterances are often associated with beliefs but also what we call *speech* *act related goals* (SARGs): SARGs reflect the intuition that people say things for a particular purpose (Asher 1999). Two rhetorical relations where beliefs and goals play a central role are: *IQAP* (Indirect Question Answer Pair) and *Q-Elab*(Question Elaboration). *IQAP*(α, β) holds if α is a question, and β together with other information the questioner has access to allows him to derive a correct answer to α (Asher and Lascarides 1998a). Note that this subsumes direct answers (so *IQAP*$(2a, 2b)$ and *QAP*$(2a, 2b)$ hold, since (2b) is a direct answer to (2a)). Second, *Q-Elab*(α, β) holds if β is a question whose answers all specify part of a plan to bring about an SARG of α (e.g., *Q-Elab*$(2b, 2c)$ holds, since the SARG of (2b) is to find a time to meet next weekend, and any answers to (2c) will reduce the search for those times) (Asher 1999). We'll shortly derive axioms for inferring *IQAP* and *Q-Elab*.

Cognitive modelling has a problem analogous to the one for computing discourse content: just as the logic for computing discourse content must be shallower than the logic of discourse content itself, the logic for cognitive modelling must be shallower than the logic of cognitive states. This is for analogous reasons: We want the cognitive modelling that's required to perform dialogue interpretation to be computable. And as we mentioned before, consistency checks are needed, because default reasoning is necessary as a result of the lack of direct access we have to other people's cognitive states. So the underlying monotonic logic must have a validity problem that's decidable at worst, or reasoning becomes uncomputable. And this means that cognitive states must be represented in a language that is strictly less expressive than first order, and it must have a more impoverished notion of validity than first order logic.

We achieve this as follows: like DICE, the language of cognitive modelling will be a modal propositional one augmented with $>$. This is much less expressive than the language which represents full cognitive states; according to Asher (1986) and Kamp (1990) this should be at least as expressive as the language of information content, and so in this case it would be SDRSs. However, the less expressive language is 'linked' to cognitive states and to discourse content, because the propositional variables in this language get indexed to the labels for SDRSs that appear at these discourse and cognitive levels. So, let π label the SDRS K_π, and let the proposition variable p_π be indexed to π. Then an interpretation of the language for cognitive modelling is *admissible* only if the worlds assigned to p_π are the ones that make K_π true. Modal operators $\mathcal{B}$ (believes) and $\mathcal{I}$ (intends) will then operate over p_π, and because of admissibility, $\mathcal{B}_A p_\pi$ corresponds to A believing the content represented in the SDRS K_π. We'll assume that whenever an agent intends something he does not already believe that it is true; in symbols, $\mathcal{I}_A\phi \rightarrow \neg\mathcal{B}_A\phi$.

We'll now provide some axioms of rationality and cooperativity. First, consider Cooperativity: an agent B is cooperative with agent A if he adopts A's goals. If B doesn't do this for whatever reason, then being cooperative means he will normally indicate this to A (Asher 1999). These principles are represented as a pair of axioms:

- `Cooperativity`
  (a) $\mathcal{I}_A\phi > \mathcal{I}_B\phi$
  (b) $(\mathcal{I}_A\phi \wedge \neg\mathcal{I}_B\phi) > \mathcal{I}_B\mathcal{B}_A\neg\mathcal{I}_B\phi$

Part (a) is clearly default: B may have conflicting goals that stop him from adopting A's goals, for example. If he doesn't adopt A's goals, then part (b) means he should make it evident to A. But this is default too: B may not, for various reasons, want A to know. For now, *pace* Sperber and Wilson (1986) who claim that cooperativity is epiphenomenal, we assume `Cooperativity` is a basic principle.

`Cooperativity` models goal transfer, but it doesn't yield an agent's goals from the evidence of what he said. In general, this is hard to do! But certain types of speech acts have goals or SARGs which one can compute in a relatively straightforward manner. It's these goals that *prima facie* we have an obligation to adopt via `Cooperativity`. For instance, questions have SARGs which will call *question related goals* (QRGs): the QRG of a question is to know an answer ($A:?\alpha$ means that A said $?\alpha$, and $?\alpha$ means α is a question).

- `Question Related Goals (QRG):`
  $(A:?\alpha \wedge QAP(\alpha,\beta)) > (\neg\mathcal{B}_A\beta \wedge \mathcal{I}_A\mathcal{B}_A\beta)$

So asking a question can lead to an inference about at least one of its SARGs.

We've so far linked goals and utterances. Grice's maxim of Quality links beliefs and utterances. It stipulates that one should say only that which one thinks is true, and avoid saying things for which one lacks adequate evidence. `Sincerity` represents the first part of this maxim (see Perrault, 1990):

- `Sincerity:` $A{:}\phi > \mathcal{B}_A\phi$

Like QRG, `Sincerity` links what one says to one's attitudes. Grice's second part to Quality provides a rule for belief transfer. In essence, it stipulates that interpreters assume by default that an agent who volunteers some information is *competent* with respect to it, i.e., what he says is true:

- `Competence:` $\mathcal{B}_A\phi > \phi$

Lewis (1969) argues persuasively that such a default must form the basis of cognitive modelling if one is to explain why linguistic conventions occur. It works as a principle of belief transfer in the following way. Suppose B hears A say that ϕ. Then by `Sincerity` B infers $\mathcal{B}_A\phi$ and by `Competence` he infers ϕ. So B must believe ϕ or succumb to Moore's (1912) paradox *ϕ but I don't believe ϕ.*

Finally, intentions, belief and action all interact. In Asher and Lascarides (1998a), we argued that Aristotle's Practical Syllogism is useful for modelling this: normally, people intend to do things that normally will eventually fulfill their goals. We encode this principle below: if (a) B intends ψ and believes $\neg\psi$ and (b) B believes that he can nonmonotonically infer ψ if ϕ is true, and ϕ is B's choice for achieving ψ, then (c) B intends ϕ:

- `Practical Syllogism (PS):`
  (a) $(\mathcal{I}_B\psi \wedge$
  (b) $\mathcal{B}_B(\phi > \mathit{eventually}(\psi)) \wedge \mathit{choice}_B(\phi,\psi))$
  (c) $> \mathcal{I}_B\phi$

We will often use `PS` and an additional axiom that actions are intentional to infer an agent's attitudes from his behaviour. For example, suppose we observe a speech act performed by A. Then we'll infer, given absence of evidence to the contrary, that A intended this action. According to `PS`, A's intention resulted from a particular mix of goals and beliefs: by fixing either one we'll abduce the other. Abduction boils down to deduction in this framework: we rearrange the antecedent and consequent of `PS` to form new axioms, but for reasons of space we don't spell out all permutations of (a), (b) and (c) here. `PS` as written above is used to infer expectations about the speech act B will perform, given some prior knowledge of his goals and beliefs.

5 Computing Discourse Structure

Recall that *IQAP*(A:?α, β) holds only if A can use β to compute a direct answer to α:

- `Axiom on IQAP:`
  IQAP(A:α, β) $\rightarrow \exists\gamma(QAP(\alpha, \gamma) \wedge \mathcal{B}_A(\beta > \gamma))$

This is a necessary condition for *IQAP*, but what are the sufficient conditions? Cognitive modelling supplies an answer.

Suppose A asks ?α. Then B uses QRG to infer that A wants to know an answer. Now, since A and B mutually believe the axioms in §4, A believes that B will deduce this. Moreover, by `Cooperativity`, B adopts A's goal and A infers B does this. So A and B both believe clause (a) of `PS` is verified, where ϕ is substituted with "A believes an answer to α". Now, A observes B utter β. By the axiom that actions are rational, A assumes this was intentional and it was B's choice for achieving ϕ. So A believes (and B believes) that clause (c) of `PS` is verified, where ψ is substituted with "B utters β". So A can abduce clause (b): B thinks β by default leads to the goal. That is, β provides A with enough information to infer an answer to α. So A infers that B believes β is attached to α with *IQAP*. Given `Competence`, A can also infer *IQAP*(α, β), as long as the constraint is met—namely, that A can indeed derive an answer to α from β. Thus the following is valid:

- `PS`, `Cooperativity`, `Competence`, $\langle\tau, ?\alpha, \beta\rangle \mathrel{|\!\!\approx}$ *IQAP*(α, β)

The consequence relation $\mathrel{|\!\!\approx}$ has a weak deduction property such that the above makes `IQAP` valid:

- `IQAP`: $\langle\tau, ?\alpha, \beta\rangle >$ *IQAP*(α, β)

Note that the only condition for this rule applying is that the surface speech act of α is a question.

Now consider *Q-Elab*. Informally, *Q-Elab*(A:α, β) holds only if (a) β is a question, and (b) for any γ such that *IQAP*(β, γ) holds (i.e., γ's an indirect answer to β), γ provides information from which A can specify a plan p for achieving the SARG of α, and this information wasn't derivable from what A and B mutually knew on the basis of the discourse so far. Given this constraint on *Q-Elab*, we can use the cognitive principles from §4 to derive an axiom for inferring *Q-Elab* at the discourse level:

- `Q-Elab`: $\langle\tau, \alpha, ?\beta\rangle >$ *Q-Elab*(α, β)

This axiom stipulates that by default, questions attach to their antecedents with *Q-Elab*. That is, the default role of a question is to help achieve a goal that prompted some previous utterance. Clarification questions (e.g., (4ab)) are a particular kind of *Q-Elab*, since one often can't achieve the SARG of α until one is clear on the content of α (in the case of (4a) the SARG is that A get to a particular location):

(4) a. A: I need to be in Cambridge tomorrow.

b. B: Cambridge UK or Cambridge USA?

b′. B: Is your hair grey?

Q-Elab also connects (2c) to (2b), because any answer to (2c) will further specify a plan to find a time to meet: an answer will either eliminate Saturday afternoon from the search (as (2d) does) or it will restrict the search to it. However, (4ab′) is predicted to be incoherent, because although the antecedent to `Q-Elab` is verified, its consequent can't be inferred because there are answers to (4b′) which don't specify plans to achieve the SARG of (4a) and therefore *Q-Elab*$(4a, 4b')$ can't be true. Since the DICE axioms give no other candidate, a rhetorical connection between (4a) and (4b′) can't be computed.

`Q-Elab` is derived from the cognitive principles as follows. Suppose A utters α with SARG ϕ. Then by `Cooperativity`, B adopts ϕ. Hence clause (a) of PS is satisfied. A observes B ask a question β, and by the rationality of action, he assumes this was intentional. So A knows that clause (c) of PS is satisfied too. So as before, A (and we) can infer clause (b): B believes that his asking β will normally lead to ϕ. Now, β has an SARG, which by QRG is that B know an answer. How does this goal fit in with B's other goal ϕ? B must construct a plan for achieving ϕ; similarly for the QRG. And by the default link given in clause (b) of PS, the plan for achieving the QRG is a subplan of that for achieving ϕ. Hence B thinks that the answers to β will specify a plan that will lead to the fulfilment of A's SARG ϕ. By `Competence`, A (and we) conclude by default that B's question does have these properties. And this is part of what the constraint on *Q-Elab* specifies. The constraint also stipulates that the relevant information isn't available to A and B already. But if it were available, then B's question would be moot and QRG wouldn't fire. So the default QRG implies that answers to questions are informative, and this yields the second part of the *Q-Elab* constraint. Hence the cognitive axioms yield `Q-Elab` as an axiom of DICE.

Having illustrated how cognitive principles lead to discourse ones, let's return to (2), and in particular the content of *2pm*. Let's assume A and B have a goal $\mathcal{G}$, which is to gain *de re* knowledge of a time at which they can meet. This yields at least two SARGs for (2a). First, there's the goal $\mathcal{G}$. (2a)'s compositional semantics are consistent with this SARG, because any answers to (2a) will either add *de re* knowledge that they can meet next weekend, or it will add knowledge that they can't. Either way, the possible times to meet are reduced. Second, by QRG, an SARG of (2a) is that A know an answer.

A and B must now attach the representation of (2b) to the representation of (2a) with a rhetorical relation. Assuming the compositional semantics of *How about* questions taken from Sag and Ginzburg (in press), answers are adjectives. Therefore, it's consistent to attach (2b) to (2a) with *IQAP*. So this is inferred via `IQAP`. Furthermore, the content of (2b) helps specify a plan to achieve the other SARG $\mathcal{G}$ of α, since it provides new *de re* knowledge that A and B can meet next weekend. A rhetorical relation *Plan-Elab*—which is analogous to *Q-Elab* but where the second utterance is an assertion instead of a question—is inferred in SDRT under these circumstances:[3] If *Plan-Elab*(α, β) holds, then β specifies a plan q to achieve the SARG of α.

What's the SARG of (2b)? Well, it's plausible that normally, a plan q which leads to the SARG of α and which was derived from a cooperative response β is the SARG of β. We can formalise this as an axiom `SARGs of Cooperative Responses`, but we forego this here. At any rate, this means that the SARG of (2b) is to accomplish a plan q, to find *de re* knowledge of a time to meet *next weekend.*

The question (2c) can attach to either (2a) or (2b) since *IQAP* and *Plan-Elab* are subordinating relations. In both cases, the default `Q-Elab` would apply. However, `Cooperativity` leads to a default that new information attaches to the lowest site in the context: if it attaches higher up, then the SARG of the lower attachment site is never addressed, but `Cooperativity` commits the participants to addressing all goals by default. Therefore, (2c) attaches to (2b) (and so the elided question resolves to *How about meeting on Saturday afternoon next weekend?*).

The discourse structure so far allows (2d) to attach to (2c), (2b) or (2a). If (2d) attaches to (2c), then *then* resolves to *Saturday afternoon*. If it attaches to (2b) or (2a), it resolves to *next weekend*. There are two principles in SDRT for preferring the attachment to (2c). First, there's the default for low attachment we just mentioned. And second, attaching (2d) to (2b/a) produces a less coherent discourse, on the grounds that A's goal for asking the question (2a) would be moot. So a general default that discourse interpretations which maximise coherence are preferred (Asher and Lascarides 1998b) means (2d) attaches to (2c). Since (2d) attaches to (2c) with *IQAP*, *then* resolves to Saturday afternoon. Moreover, the plan q which this answer to (2c) gives rise to is to find a time to meet which is next weekend, but not on Saturday afternoon. By `SARGs of Cooperative Responses`, q is the SARG for (2d).

We now come to (2e). This can attach to (2d) or (2a). There are two reasons why it doesn't attach to (2a): first, the default for low attachment blocks it; and second, one can't resolve the anaphoric expression *2pm* with this attachment, because *next weekend* provides two alternative bridging relations to *2pm*, violating the constraint on uniqueness given by Clark (1977). So (2e) must attach to (2d). This means that the only available antecedent that *2pm* can resolve to is Saturday afternoon. The antecedent to `Q-Elab` is verified. However, its consequent can't be inferred because the constraints on *Q-Elab* are violated. This is because given that (2e) means *How about 2pm on Saturday?*, answers to this question fail to specify a plan to know of a meeting time that's *not* on Saturday afternoon. But given the SARG of (2d), (2e) must provide such a plan if it attaches with *Q-Elab*. And so (2e) can't attach to (2d) with *Q-Elab*. But no other default rules are verified. So no relation can be inferred for attaching (2e), resulting in incoherence.

6 Conclusion

Principles of rationality and cooperativity can account within SDRT for some of the links between cognitive and conventional information that are needed in an adequate account of anaphora in dialogue. We used the cognitive principles to derive axioms for inferring rhetorical relations at the discourse level and to give these relations a semantics that constrains the resolution of anaphora.

The cognitive axioms ensure that *IQAP* is the default way of attaching an assertion to a question, and questions are attached by default with *Q-Elab*. These defaults follow from `Cooperativity`: participants engage in a cooperative effort to solve the SARGs of utterances. These defaults also represent expectations about responses.

But in some situations an agent doesn't give the expected response. For instance, `Cooperativity` may not apply, perhaps because B believes that A's SARG is already achieved, or he believes that there isn't any way of achieving the SARG. These exceptions arise from constraints on intentions adopted by Cohen and Levesque (1990), Koons and Asher (1994) and others. If B doesn't adopt A's SARG, then by part (b) of `Cooperativity` he is by default obliged to indicate this. So in these circumstances, B's question or assertion may have a different rhetorical role: instead of providing a *Q-Elab* or *IQAP*, it may be a contribution which is designed to imply that the SARG isn't adopted. In future work, we plan to explore such exceptions, and how one can use clues in the dialogue to infer them. We also plan to analyse indirect speech acts, and explain why utterances with similar meaning can have distinct SARGS; e.g., *Can you pass the salt?* vs. *Do you have the physical ability to pass the salt?*.

Notes

[1]This mapping is independently motivated on the grounds of VP ellipsis, gapping, and other ellided constructions.

[2]We introduced links between cognitive states and discourse structure in Asher and Lascarides 1994, and Asher and Lascarides 1998a but the modules were not distinguished.

[3]In fact, one can use the axioms in §4 to derive an axiom for inferring *Plan-Elab*, but we gloss over this here.

7 References

Asher, N. [1986] Belief in Discourse Representation Theory, *Journal of Philosophical Logic*, **15**, pp.127–189.

Asher, N. [1993] *Reference to Abstract Objects in Discourse*, Kluwer Academic Publishers.

Asher, N. [1999] Discourse Structure and the Logic of Conversation. *Current Research in the Semantics Pragmatics Interface.*

Asher, N. and A. Lascarides [1994] Intentions and Information in Discourse, in *Proceedings of ACL94*, pp.35–41.

Asher, N. and A. Lascarides [1998a] Questions in Dialogue, *Linguistics and Philosophy*, **23.3**, pp237–309.

Asher, N. and A. Lascarides [1998b] The Semantics and Pragmatics of Presupposition, *Journal of Semantics*, **15**, pp239–299, OUP.

Clark, H. [1977] Bridging, in P. N. Johnson-Laird, P. C. Wason (eds.) *Thinking: Readings in Cognitive Science*, pp. 411–420, CUP.

Cohen, P. and Levesque, H. [1990] Rational Interaction as the Basis for Communication, in in P. R. Cohen, J. Morgan and M. Pollack (eds.) *Intentions in Communication*, pp.221–255, MIT Press.

Grice, H. P [1975] Logic and Conversation, in Cole, Peter and Morgan, Jerry L. (eds.) *Syntax and Semantics Volume 3: Speech Acts*, pp41-58. Academic Press.

Groenendijk, J. and M. Stokhof [1991] Dynamic Predicate Logic, *Linguistics and Philosophy*, **14**, pp39–100.

Grosz, B. and Sidner, C. [1986] Attention, Intentions and the Structure of Discourse. *Computational Linguistics*, **12**, 175–204.

Grosz, B. and Sidner, C. [1990] Plans for Discourse, in P. R. Cohen, J. Morgan and M. Pollack (eds.) *Intentions in Communication*, pp.417–444, MIT Press.

Hobbs, J. R., M. Stickel, D. Appelt and P. Martin [1993], Interpretation as Abduction, *Artificial Intelligence*, **63 (1–2)**, pp.69–142.

Kamp, H. [1990] Propositional Attitudes, in C. A. Anderson, J. Owens (eds.), *The Role of Content in Logic, Language, and Mind*, pp.27–90, CSLI.

Kamp, H. and Reyle, U. [1993] *From Discourse to Logic*, Kluwer Academic Publishers.

Koons, R. C. and N. Asher [1994] The Revision of Beliefs in a Changing World, *Proceedings of TARK5*, pp. 321-340.

Lascarides, A. and N. Asher [1993] Temporal Interpretation, Discourse Relations and Commonsense Entailment, in *Linguistics and Philosophy*, **16**, pp.437–493.

Lewis, D. [1969] *Convention: A Philosophical Study*, Blackwells.

Litman, D. and J. Allen [1990] Discourse Processing and Commonsense Plans, in P. R. Cohen, J. Morgan and M. Pollack (eds.) *Intentions in Communication*, pp.365--388, MIT Press.

Moore, G.E. [1912] *Ethics*, London.

Moore, J. D. and C. Paris [1993] Planning Text for Advisory Dialogues, *Computational Linguistics*, **19.4**, pp.651–694.

Perrault, R. [1990] An Application of Default Logic to Speech Act Theory, in P. R. Cohen, J. Morgan and M. Pollack (eds.) *Intentions in Communication*, pp.161-186, MIT Press.

Sag, I. A., and Ginzburg, J. [in press] *Interrogatives*, CSLI publications.

Sparck Jones, K. [1989] Realism about User Modelling, in Kobsa, A. and W. Wahlster (eds.) *User Models in Dialog Systems*, pp341–363, Springer-Verlag.

Sperber, D. and Wilson, D. [1986] *Relevance*, Blackwells.

Alex Lascarides
Division of Informatics,
University of Edinburgh,
2, Buccleuch Place,
Edinburgh, EH8 9LW,
Scotland, UK
alex@cogsci.ed.ac.uk

Nicholas Asher
Department of Philosophy,
University of Texas at Austin,
316 Waggener Hall,
Austin,
Texas, 78712,
USA
nasher@bertie.la.utexas.edu

Towards a Robust Semantics for Dialogue using Flat Structures

David Milward
SRI International

Dialogue utterances are rarely sentences and are often fragmentary. This paper discusses some of the implicit assumptions made by shallow and deep processing approaches, and advocates a new approach which keeps the robustness of shallow or keyword-based approaches, whilst retaining the generality and formal nature of a full semantics.

1. Introduction

The ideal spoken dialogue system should be flexible, allowing users to supply extra information from that specifically asked for, or to take the initiative. However, these aims can be difficult to square with the need for top down expectation to help speech recognition accuracy. Thus there tends to be a divide between tightly constrained systems with e.g. separate finite state grammars to recognise uses responses to particular system queries, and unconstrained systems which rely on keyword spotting/pattern matching, or attempt deep language analysis. In this paper we will try to bridge various divides between shallow and deep processing systems and show how top down expectation from dialogue context can still be incorporated in a flexible system. The key to the approach is to use a 'flat' representation where information about the semantic (or syntactic) structure is distributed between a set of constraints. This makes it possible to combine the robustness we would expect from pattern matching without wasting any available linguistic information concerning constituent structure.

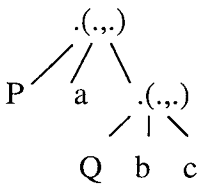
The paper is split into four main parts. Section 2 describes flat structures. Section 3 describes various requirements and issues in dialogue processing. Section 4 describes an approach to dialogue processing using flat structures. Section 5 evaluates the approach according to the requirements described in Section 3. Section 6 describes a preliminary implementation.

2. Introduction to Flat Structures

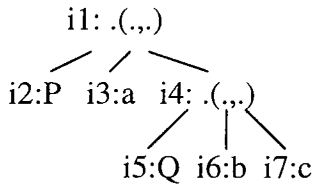
The most basic form of flat structure is just an indexed version of a standard semantic representation. Consider a standard recursively structured piece of semantic representation:

$$P(a,Q(b,c))$$

This can be represented as the application structure:



Now consider giving each item in the structure a unique label i.e.



This information can be represented by the set of constraints:

i1:i2(i3,i4), i2:P, i3:a, i4:i5(i6,i7), i5:Q, i6:b, i7:c

The resulting representation is flat in the sense that there is no explicit recursion. The set of constraints, i2:P, i3:a, i5:Q, i6:b, i7:c describe the lexical/morphological content of the representation. The constraints i1:i2(i3,i4) and i4:i5(i6,i7) describe the structural aspects.
For example, consider the representation we would might get for the sentence:

John believes Jack runs

The following flat structure corresponds to the logical form, believe(john,run(jack)):

i1:i2(i3,i4), i2:believe, i3:john, i4:i5(i6), i5:run, i6:jack

This provides the morphemes, believe, john, run and jack along with constraints specifying their relationship.

Note that we have only changed how we represent the semantics: there is a one-to-one mapping between the set of constraints and the original recusive representation (assuming index renumbering). The basic flat structure can be regarded as an alternative notation, or as a description of the original semantic representation.

It should be noted that there are many different varieties of indexed/flat structures, going back at least to Kay 1970. For example, neo-Davidsonian semantics is sometimes described as a flat representation, since event variables act somewhat similarly to indices. The semantics for a sentence such as "John runs at 5pm" is given by a conjunction of two constraints hanging off an event variable i.e.

$\exists$e. run(e,'j) & at(e,5)

This enables inferences such as "John runs at 5pm" therefore "John runs" to go through without the need for meaning postulates. Hobbs (1983) extended this approach to all predicates by mapping each n-ary predicate to an n+1-ary predicate including an event variable, thus allowing restrictive modification to be done via conjunctive constraints.

A rather different kind of flat structure (closer to the basic flat structures described above) has been used as a way to provide semantics for fragments which do not form standard constituents (Milward 1991). For example, the semantics of "Mary Fred" in the sentence "John showed Mary Fred or Peter Sue" is treated as the set of constraints, {i4:mary, i5:fred}. An intermediate semantic representation (prior to quantifier and conjunction scoping) uses conjunction or disjunction of sets of constraints e.g. OR({i4:mary, i5:fred}{i4:peter,i5:sue}).

More recently, flat structures which combine both a Davidsonian approach to modification with indexing similar to basic flat structures have been used in Machine Translation (Copestake et al. 1995) following the lexicalist approach of Whitelock (1992) and Trujillo (1995). The prime motivation within Machine Translation is that we can more easily express translation rules which pick up disconnected pieces of the source semantic representation and map them to a single piece of the target representation (or vice versa). Transfer between source language and target language representations is achieved by mapping between a subset of the conjuncts in the source to another set of conjuncts in the target. The translation process is inherently bottom up: we translate subset by subset, and the final set of target conjuncts should comprise a fully connected semantic representation.

Finally, there has also been much interest in using indexed representations for underspecified semantic representation (e.g. Reyle 1993, Egg 1998). Here the emphasis has mainly been on weakening structural constraints to enable underspecified representation of quantifier scope. Structural constraints are divided into dominance, precedence and immediate dominance constraints (similar to the work of Marcus et al. 1983 on the description of syntactic tree structure) making it possible to state that a piece of representation is within the scope of another, without further specifying the relationship.

The interest in flat structures here is motivated by the distributed nature of the representation. There is thus a better chance of achieving robustness when dealing with fragmentary input, or in making rules (e.g. mapping to database slot-values) sensitive to other parts of a representation. To illustrate the former, consider a case where a structural analysis has produced three separate fragments P, a, and Q(b,c). This information can be represented by the set of constraints:

i2:P, i3:a, i4:i5(i6,i7), i5:Q, i6:b, i7:c

This is a subset of the flat structured representations for P(a,Q(b,c)), with only the constraint, i1:i2(i3,i4) missing. The change of representations thus has the advantage of making the semantics of fragments and full constituents look more similar. We will see that this in turn suggests ways in which rules (e.g. for mapping to database slot-values) can be relaxed to ensure robustness.

3. Dialogue Systems

3.1 Current Systems

Flexible dialogue systems tend to be either based on shallow processing (e.g. keyword spotting or pattern matching) or deep processing (interpretation down to a level of logical or quasi logical form). Shallow systems are task directed, and do not attempt a full semantic or pragmatic analysis of all parts of an utterance, although processing may be reasonably detailed for the parts relevant to the task. Deep processing systems tend to have a layered architecture e.g.

spoken input
↓ *recogniser*

n-best list / word lattice
↓ *syntactic parse/ semantic analysis*

candidate logical form/ quasi-logical form
↓ *translation*

database slot possibilities
↓ *dialogue manager*

integration with current context

This is the basis of the architecture of the deep processing component of the SLT system (Boye et al., 1999). The SLT system keeps the top 5 analyses produced by the recogniser to allow later stages of processing to act as a filter on the results of the recogniser. Statistically trained triples (Carter 1997) build in domain dependence to choose the best syntactic/semantic parse. Other systems such as Verbmobil (Kasper et al. 1999, Goertz et al. 1999) and OVIS (van Noord et al, 1999) keep the recogniser lattice, annotating this with syntactic and semantic information.

There is often an assumption that deep approaches will provide better results than shallow approaches in the long term. All we need is more work on our grammars and faster machines. In the meantime we should use multi-engined approaches where you use full analysis wherever possible, but back off to shallow analysis where deep analysis fails (either through time out or lack of coverage). This kind of argument was very much the inspiration for the SLT system and it assumes that when deep analysis provides a result it will generally be better than the result of shallow processing. However, empirical evaluation of the resulting system gave exactly the opposite result. Lewin et. al. (1999) report that when the results of the two analysis modes disagreed, the shallow analysis provided a better result three times as often. One of the inspirations for this paper was to try to see if there might be some theoretical reasons for these results, so let us now consider how various issues tend to be handled by the two kinds of systems.

3.2 Requirements for Dialogue Systems

Robustness

Shallow and deep systems tend to tackle the issue of robustness differently. Many shallow approaches home in on words or phrases which are relevant to the task. This means that unknown vocabulary or unknown grammatical constructions will normally have little or no affect on the processing. To give an example, a keyword based approach which only looks for city names should easily deal with an interchange such as:

Where do you want to go?

Well let me see now, um, well I think Boston, yes that's right

In contrast, a deep approach will normally come up with a set of possibly overlapping analysed fragments. There are then two further tasks. The first is to pick the best set of fragments. The second is how to translate the fragments into some kind of database update function (or whatever is required for the task). A common approach to the first task (e.g. Verbmobil) is to prefer the shortest possible path though a set of fragments i.e. the smallest number of fragments which span the utterance. When a single constituent spans the whole utterance this will be picked since the path is just of length 1. No contextual information is included at this stage. For the second task, there are two common approaches. The first is to translate the fragments independently into database update functions. The second is to apply various reconstruction rules (working on either syntactic or semantic fragments) to try to fit the pieces into a full sentence (c.f. Verbmobil).

Domain Dependence

Consider the following sentence in the Air Traffic Domain:

Show flights to Boston

This has two readings, one where "flights to Boston" is a constituent, the other where "to Boston" is an adverbial modifier (similar to "to Fred" in "Show flights to Fred").

A shallow system may well have rules for "flights to <city>", but is unlikely to include rules for the adverbial reading. Thus the possibility of ambiguity does not exist. Despite there being no explicit domain training, the correct reading is picked.

In contrast, in a deep analysis system with a general purpose grammar, there has to be specific customisation to the domain. This may be via specialisation of the grammar to the domain (c.f. OVIS), or via the introduction of domain specific preference mechanisms. For example, the SLT system uses 'treebanking' (Carter 1997) which involves a human picking correct analyses in order for the machine to learn domain specific syntactic and semantic triples which select between alternative readings.

Other examples may be problematic for shallow systems which employ domain independent chunking. For example, consider the following interchange:

Please give me your departure and destination cities

Boston London Heathrow

The correct bracketing here requires the use of domain specific information that "London Heathrow" can be treated as a unit. To use this information we either need to incorporate a domain specific pre-processing stage which picks out idiomatic expressions, or preserve alternative bracketings until later domain specific processing.

Finally we should note that domain dependence (and context dependence as we shall discuss below) should not just be used for choosing between readings given by full parses of the utterance, but should affect which fragments are picked. Sometimes this will mean that a fragment interpretation should be chosen instead of a full interpretation. For example, a relatively common problem with the current SLT system is where the recogniser suggests an extra word at the end of an utterance. A full analysis can often be found which incorrectly incorporates the bogus word, and this is picked in preference to a correct fragment analysis.

Context Dependence: Previous Utterance

Shallow systems typically incorporate preferences arising from the preceding question. This is used to ensure that the answer "Boston" in the context of the question "Where do you want to go" is interpreted differently from in the context of "Where do you want to leave from". In deep systems the need for context dependent preferences is not so immediately obvious, since examples such as these can be treated using some variety of ellipsis resolution where the utterance is first translated into something equivalent to "I want to go to Boston". This however breaks down in cases where there is a full sentence reply. Consider the following example (which occurred in the SLT system). The interchange is as follows:

S: Where would you like to go?

U: I would like to go to/from Boston

The speech recogniser fails to distinguish between to/from, and domain specific preferences happen to very slightly prefer “from” vs . “to” in this context. The system thus decides (incorrectly) that the most likely sentence is "I would like to go from Boston", The correct analysis has now been ruled out, and the dialogue manager cannot recover the correct interpretation.

How can this be changed? There are two options. The first is to bring contextual information to bear much earlier in the process (this is effectively what is happening in the shallow approaches). The second is to ensure all analyses survive until a late stage, where context then comes into play. In the SLT system domain specific preferences could be context specific as well, though this would entail a larger treebanking stage. In OVIS there has already been some experimentation with bringing in context early in the process and weighting fragments accordingly.

Context Dependence: Other parts of the utterance

Consider the following examples:

I'd like to leave York, now let's see, yes, at 3pm

at 3pm ⇒ departure_time(3pm)

I'd like to arrive at York, now let's see, yes, at 3 pm

at 3pm ⇒ arrival_time(3pm)

The translation of the phrase "at 3pm" is dependent here not on any outside context but on the rest of the utterance. This is relatively easy to cope with in a shallow processing system which works on a single analysis at a time. We merely need patterns of the form:

[leave] .... [at 3pm]/time(3pm) $\Rightarrow$ departure_time = 3pm
[arrive] .... [at 3pm]/time(3pm) $\Rightarrow$ arrival_time = 3pm

There is no a priori reason to suggest that we could not apply similar context dependent rules within a deep approach. although it might be necessary to apply incomplete heuristics to avoid the inefficiency caused by fragments not being able to be translated independently.

Reconfigurability

It is sometimes argued that deep approaches are more easily reconfigurable since there is more reuse of standard processing modules (e.g. parsing, morphology and semantic construction). Shallow processing systems tend to be more task directed, so we might expect a greater proportion of the code to need changing. However in practice this argument only tends to apply when there is a change of task (e.g. enquiries regarding routes vs. prices), but no change of domain (e.g. computer manuals vs. airline booking). When moving to a new domain a deep approach has more work to do since it doesn't just have to deal with words or phrases of particular interest to the task, but all new words and constructions found in the domain.

Where shallow systems tend to be more problematic is when we try to improve coverage. Many systems rely on precisely ordered pattern matching rules. Adding a rule or changing the order for one phenomenon often causes another to break. This is not unlike trying to maintain a large set of grammar rules in a deep analysis system.

Accuracy

The final issue to consider is the accuracy of analyses produced by shallow or deep systems. A feature of most shallow systems is that they are goal directed. Similar to Information Extraction (IE) Systems, they only look for the information they need and ignore the rest. Let us look at an example from IE first, where the issues are particularly clear. A standard task in IE is to look for relationships between people and companies e.g. "who is chairman of which company". Consider the following sentence:

> John Smith, Director of ABC Corp., announced today that his four task force managers had increased profits in three divisions.

From this we can infer that John Smith is Director of ABC Corp. without unpacking all the readings for the sentence as a whole. This is a valid inference to make since the sentence can be rephrased as the following conjunction.

John Smith is Director of ABC Corp. and John Smith announced ...

It is safe to infer "A" from "A and B" even if "B" is ambiguous.

Another standard abstraction made in IE is to ignore adjectives or other noun modifiers and just look at the head noun. For example, a system might extract "ABC Corp. made a profit" from all three sentences below, although it is only safe to do so from the first:

ABC Corp. announced a pre-tax profit of $4000 dollars
ABC Corp. falsely announced a pre-tax profit of $4000 dollars
ABC Corp. announced a false pre-tax profit of $4000 dollars

Dialogue is a bit different since we don't usually have to wade through immense quantities of irrelevant text. However, there are plenty of cases where information is supplied which there is no need to process. For example in the flight booking domain we get cases such as:

I would like a comfortable flight to Boston because my sister .....

Here again we can ignore the reason, and the modifier, "comfortable", to extract the relevant request of "I would like a flight to Boston".

There are also similar cases in dialogue where it is not safe to ignore surrounding material e.g. it is safe to use shallow processing to pick out "destination = Boston" in the first example below, but not the second

I would like to go to Boston
Now let me see, not to Boston, but perhaps to Chicago

How can we deal with this kind of negative information? The first option is just to ignore it. Negation (implicit or explicit) and non-intersective modification is not that common in newspaper texts or in many dialogue scenarios, so the correct inferences will normally go through. However, this means we are unnecessarily throwing away useful information. The second option is to perform some checking of the monotonicity properties of the context e.g. by checking there is no 'not' with the potential to have scope over the material of interest. The third option is to move closer to deep processing, since part of the job of full parsing is to determine constituency, hence scoping, and the job of a grammar to provide useful generalisations using the recursive structure of a language.

3.3 Summary

What conclusions can we make from the discussion above? Firstly, the shallow processing paradigm does have theoretical justification. In shallow processing you make (defeasible) inferences using partial information about the utterance content. This is similar to how Oaksford and Chater (1991) argue that humans generally have to behave: we often have to make defeasible inferences from partial information. Even if we know what utterance was made we may not know whether it was made sarcastically or not. At some stage we have to jump to a conclusion, although this may be subject to later retraction.

Secondly, although shallow systems are good at dealing with partial information, in some cases (e.g. the discussion of negation in the last section), the information they use may be more partial than it needs to be. Where we can correctly ascertain constituency (and hence scope) we should do so.

4. An Approach using Flat Structures

4.1 Choice of Flat Structure

Shallow approaches seem to gain by concentrating on information which is of interest to a specific task. To be able to do this in a deeper approach it is useful to use a representation which splits information up as much as possible. Here we will use a basic flat structure as described in Section 2 but make two extensions. The first is to allow for the same index to be used more than once, and to treat such cases as alternative readings (i.e. meta-level disjunction). For example, we take i4:P i4:Q to mean i4 has the two readings, P and Q[1]. We will also allow the same index to appear in more than one argument position (for example, the following would be a valid set of constraints: {i4:(i1,i2), i4(i1,i3)}). These two extensions give us the ability to pack ambiguous readings in a similar manner to a chart, and to structure share similar to a chart or packed parse forest.

4.2 Building a Flat Structure

The extensions made above give us something akin to a 'semantic' chart, and allow flat semantic representation to be created directly during parsing for both fragments and full constituents. The choice to give indices names such as i1, i2 etc. was arbitrary, so we can equally well choose to name them e.g. 0-1-np (corresponding to a np edge between positions 0 and 1) or 1-2-vp (corresponding to a vp edge between positions 1 and 2). We merely have to ensure that if indices are used more than once, then their interpretation corresponds to the choices made above. This will be the case if we choose chart edges with the appropriate syntactic information and have a close syntax-semantics mapping. Semantics thus becomes a further annotation on the chart. Consider the syntactic chart which might be created for the sentence "US205 leave Boston":

0-1-np: US205
1-2-vtr: leave
2-3-np: Boston
1-3-vp: [1-2-vtr,2-3-np]
0-3-s: [0-1-np,1-3-vp]

By further annotating the edges we can produce a corresponding 'semantic' chart i.e.

0-1-np: us205
1-2-vtr: leave
2-3-np: boston
0-3-s: 1-2-vtr(0-1-np,2-3-np)

---

[1] Note that in Minimal Recursion Semantic (Copestake et al. 1995) there is a similar ability to use the same index more than once, but the interpretation is rather different. In MRS, i4:P, i4:Q is equivalent to conjoining Pand Q.

The edge left out is the vp edge. We'll choose to treat the semantics of the vp in a slightly unconventional way, similar to the semantics for the sentence, but with a non existing np index i.e.

1-3-vp: 1-2-vtr(1-1-np,2-3-np)

Similarly, assuming a bottom-up, left-corner or incremental parsing strategy, the fragment "leaves Boston" would get the semantic chart:

0-1-vtr: leave
1-2-np: boston
0-2-vp: 0-1-vtr(0-0-np,1-2-np)

We can think of the vp semantics as a combination of the vp with an unknown empty np. This seems to work well: we cannot infer anything about the subject of the verb but we retrain the correct relationship between the verb and its object (allowing, for example, a rule to infer that "boston" is the departure city).

Note that although the positions used here i.e. 0,1,2,3 correspond to word counts, we can just as easily use positions corresponding to time intervals, building the semantic chart directly on top of a word lattice, as is done in Verbmobil (Worm 1999), where chart edges are similarly annotated with semantic material.

4.3 Applying rules to a semantic chart

What kind of rules might we apply?

Despite the equivalence between basic flat structures and corresponding logical forms, the different representations suggest rather different translation rules. Reconsider the example we had before of a pattern matching rule to interpret "at 3pm" in the context of leaving:

[leave] .... [at 3pm]/time(3pm) => departure_time = 3pm

This rule assumes some prior application of rules which determine sortal information i.e. that "at 3pm" is a time expression.

Now consider the chart/lattice we might get for the utterance "I leave at 3pm":

0-1-np: I
1-2-vintr: leave
0-2-s: 1-2-vintr(0-1-np)
2-3-p: at
3-4-np: 3pm
2-4-pp: 2-3-p(2-2-s,3-4-np)
0-4-s: 2-3-p(0-2-s,3-4-np)

Here we have three larger edges, the first corresponding to "I leave", the second being the pp correspondin to "at 3pm", the third corresponding to the whole sentence "I leave at 3pm".

The semantics for "at" takes the sentence as the first argument, the time as the second. The pp arc, similar to the vp arc in the previous example includes a null first argument.

Before advocating a particular approach, let us consider more generally the kind of rules which can be applied to a semantic chart. For example, a specific rule, requiring a connected semantics for the verb phrase might be as follows (the capitalised letters stand for variables):

> If we find a preposition "at" and this immediately dominates the verb "leave" then departure-time is the second argument of "at" i.e.
>
> I : J(K,L) & J:at & K: M(N) & M:leave & L:T
> =>
> departure-time = T

This rule is probably too specific, so we may want to loosen it to allow for e.g. other modifiers between the verb and the "at" phrase i.e.

> If we find a preposition "at" and this dominates the verb "leave" then departure-time is the second argument of "at" i.e.
>
> I : J(K,L) & J:at & K > H & H: M(N) & M:leave & L:T
> =>
> departure-time = T

Weakening this again we might get rid of the dominance restriction. We then have:

> If we find a preposition "at" and a verb "leave" then departure-time is the second argument of "at" i.e.
>
> I : J(K,L) & J:at & M:leave & L:T
> =>
> departure-time = T

This final rule is actually weaker than the pattern matching rule allowing "leave" to appear before or after "at", and there is also no check that the argument to "at" is of sort "time". We may want to strengthen the rule with at least a sortal check i.e.

> I : J(K,L) & J:at & M:leave & L:T & time(T)
> =>
> departure-time = T

We now have suggested several rules, the first being closest to a full semantics approach, the bottom closest to a pattern matching approach. Is one of these the correct rule?

The top rule is the most restrictive, requiring a fully connected verb phrase. The final rule is the least restrictive and thereby the most robust when encountering fragmentary input. However it is also the most likely to go wrong. For example, it would provide "departure-time = 3pm" from the sentence

> I would like to leave Cambridge and arrive at 3pm

The rule thus only makes sense in the context of other rules which can override it., e.g. a more specific "arrival-time" rule which worked on the verb phrase "arrive at 3pm". In pattern matching systems this would typically done via temporal ordering: more specific rules are applied first and block weaker rules from applying.

This discussion suggests that no one rule should be chosen. Instead we need a range of rules for "departure-time" and "arrival-time" some of which are more specific than others. In a particular scenario we should pick the most specific rule which can apply.

To compact up the rules, the approach taken here is to provide a set of obligatory constraints, and some optional constraints. The four rules above are compacted into the single rule:

> Obligatory constraints: {I : J(K,L), J:at, M:leave, L:T, time(T)}
>
> Optional constraints: {K > H, H: M(N), K : M(N)}
> =>
> departure-time = T

A particular application of a rule will get a weight according to the number of obligatory and optional constraints which hold.

Inference, Underspecification and Translation

The rules above are expressed as inference rules i.e. "if we have A and B then we can infer C". This suggests that getting the value for a slot is a process of inference from a semantic representation which may be underspecified (the rules do not check that the representation is fully specified). The inference is also defeasible, since there is no checking of the monotonicity properties of the surrounding context, and various assumptions are missing. Another way to think about this is in terms of Abduction (c.f. Hobbs et al. 1993) An inference could go through if we added in various extra constraints, and we want to add in as few extra constraints as possible (hence more specific rules are preferred). Note that this is a non-standard use of Abductive Reasoning since we are allowing abduction of information not just about the context etc. but also about the actual structure or content of the utterance.

A defeasible inference perspective works well when we consider single slot values. However, when we consider multiple values, or items in the scope of other operators, it is useful to think in terms of translation. The idea of treating database enquiry as translation goes back to at least Bronneberg et al. (1980), and the use of inference for translation into database slots was advocated by Rayner (1993). Taking a translation approach, we can consider translating individual pieces of the utterance, and then put the pieces back together again. This should allow better generalisation. It also gives us a better story as to why a more specific rule for e.g. arrival time would block a less specific rule for departure time (even if higher weighted, or applied earlier). Once a term has been translated, it will not be available for translation using another rule. To rephrase the rules above as translation rules we have to distinguish between items to be translated and the surrounding context. For example, the departure-time rule would become:

> Constraints for translated terms: {time(T), J:at, L:T, I:J(K,L)}
> Constraints for rest of utterance: {M:leave}

Optional constraints:	{ K > H, H: M(N), K : M(N)}
New constraints:	{P: translation({J,L},Q), Q:departure_time=T}

Here the two indexed items, J and L (corresponding to the word "at" and the time) are translated to the term Q which is a statement that the departure time has the value T. Other items such as the M (corresponding to the word "leave") are constraints on the rule, but are not part of the translation.

Now consider an example containing a negation e.g.

Not to London on Thursday, to Manchester on Friday

The negation can be translated separately e.g.

Constraints for translated terms:	{I:J(K,L), J:not}
Constraints for rest of utterance:	{ }
Optional constraints:	{ }
New constraint :	{R: in-order(-M,+N), P:translation(K,M), Q:translation(L,N)}

This states that there is a translation of "not" as a database command which first checks that the constraint corresponding to M does not hold, then asserts the constraint corresponding to N (note that this is just one of the many possible translations of "not" ; alternative translations are required for "not via London", or "not Boston, London").

This rule is appropriate if the material dominated by the "not" has a connected parse, with the immediately dominated material having a translation. We can weaken this to allow dominance rather than immediate dominance or backoff further e.g. to allow the first argument of "not" to be anything upto the next comma[2].

How does this translation approach differ from the defeasible inference approach? In this approach we translate all the terms we think are important, and assume that all other terms are semantically empty (null constraints or identity functions). It is this assumption that is defeasible, along with relaxations allowed within the rules.

For completeness, it should be mentioned that it would be possible to allow rules which access the values of translated subconstituents i.e. a rule may fire if subconstituents have appropriate translations. This however does introduce rule order dependence: we would have something very similar to cascaded pattern matching, where first items are annotated as e.g. companies and losses, then a rule picks up the companies and losses to annotate the sentence as a whole as a company loss.

Finally we should note that things get more complex when we consider alternative readings for the same section of utterance (which of course is the normal state of affairs when we start with a speech lattice). In this case we cannot be sure of the surrounding context, and we do not want translations of different parts of the utterance to depend upon different readings of

[2] This kind of constraint is actually a syntactic constraint, not a constraint about the content or structure of the semantic representation However, there seems no problem with mixing syntactic and semantic constraints in this kind of approach where everything can be seen as information attached to the same chart.

the context. This means that assumptions need to be preserved (e.g. the contextual constraints above), and if there are cascaded rules, the assumed constraints need to filter into the result (in logical terms this is just a case of A -> B and B -> C then A -> C). We will consider one possible architecture at the end of this paper, but this is by no means the only possibility.

Translation Rules for a Semantic Chart

The final change we need to make to the rules is to allow for the effect of the context of the previous utterance, and the dialogue contribution so far (e.g. what slots have already been filled etc.). The translation rules are thus partly dealing with semantic information, partly with what might traditionally be treated as pragmatic information. An example translation rule is the following:

Constraints for translated terms:	{time(T), J:at, L:T, I:J(K,L)}
Constraints for rest of utterance:	{ }
Constraints from prior utterance:	{M:query(departure_time)}
Contextual slot values:	_
Optional constraints:	{ }
New constraints:	{P: translation({J,L},Q), Q:departure_time=T}

This is appropriate for picking out the departure time where the question asked what the departure time was, and the answer contains a time. This rule will fire provided there is no more specific rule to translate the terms to something else.

Now let us reconsider the various requirements we had earlier and see how this approach fits.

5. Evaluation of the Flat Structure Approach

Robustness

The approach outlined above has most similarity in dealing with fragmentary input to pattern matching approaches. There is no need to try to reconstruct a full semantic analysis, instead a weaker version of the translation rule is used (with fewer optional constraints applying). Where there is ambiguity, more specific rules are preferred.

Domain Dependence

Domain dependence is built into the translations rules, rather than there being any particular filtering stage based on domain specific preferences. Different readings are preserved until the translation rules fire. Although a sentence such as "Show flights to Boston" will get two readings, there is no need to choose between them. The translation rules (similar to pattern matching rules) will only pick up on the appropriate possibility. More empirical research is required to see if this will work in all cases, or whether there might sometimes be a role for domain specific preferences to adjust the weightings given to different components.

The preservation of different readings until translation allows correct treatment of the "Boston London Heathrow" case. Finally, we should note that there is no a priori preference for longest fragments or the shortest path through the chart. Preference is given to more specific instantiations of rules. Larger fragments may well result in more optional constraints being satisfied. However this will only be if the larger fragment introduces more material which is relevant to the task: the addition of a bogus word will not be rewarded. Moreover, specificity gained in this way still has to be balanced against specificity gained from contextual constraints.

Context Dependence: Previous Utterance

Contextual dependence is included by adding constraints to the translation rules. This can comprise constraints on the previous utterance (e.g. which slots are being queried), or on the current dialogue state. In simple task oriented dialogues the state is likely to be just a set of slots, some of which are filled. Contextual constraints count towards making the rule more specific, though to a lesser extent than information within the utterance itself. The aim is for the top down expectation provided by contextual constraints to be able to affect the interpretation without overriding what the utterance actually says.

Context Dependence: Other parts of the utterance

The approach here is similar to shallow processing in allowing the translation of one part of an utterance to be affected by the translation of other parts.

Reconfigurability

Reconfiguring to a new task requires the introduction of new translation rules, and the addition of lexical entries for at least the words mentioned in the translation rules. The robust nature of the approach means that we can provide a working system without providing a full parse for all, or even for a majority of the utterances in the domain.

Accuracy

The aim of this approach was to give the robustness you might expect from shallow approaches, but use as much linguistic information as available to ensure accuracy is as good as possible. The use of rules which can be more or less specific should achieve this. Where parsing provides appropriate constituency (and hence appropriate scope for operators such as "not") this information is used. Where not, the relaxed versions of the rules should at least equal the results of shallow processing.

6. A preliminary implementation

Most of the ideas in this paper have been integrated into a prototype dialogue system designed for route planning. The system allows for simple question answer dialogues such as the following:

S: Where would you like to go?
U: London
S: Where are you leaving from?
U: Cambridge
S: Would you like the quickest or the shortest route?
U: shortest
S: When would you like to arrive?
U: Before five p m.

However, it also allows for a user to supply more information than expected or different information. For example, the following exchange is more appropriate for an experienced user of the system:.

S: Where do you want to go?
U: I would like the shortest route from Cambridge to London to arrive before 5

The system builds a chart using a fully incremental parser based which adds new edges or annotations to extend a word lattice, Categorial Grammar was used to give a straightforward syntax/semantics interface, and the grammar was subsequently compiled into simple Dependency Grammar. This enabled the use of a packed parser based on the packed incremental recogniser described by Milward (1994).

The present algorithm applies each translation rule to the chart, and picks up a set of potential slot-value pairs (for the slots, destination, source, mode, and arrival/departure time). Each slot-value pair is weighted according to the specificity of the rule used. The weighting is pretty crude, and achieved by adding the number of items mentioned in a rule, with higher weightings for closer items. More advanced weighting schemes based on Information Retrieval technology (where more interesting terms get higher weights) can be imagined. When spoken input is used we would also want to use the recogniser weightings.

After creating the set of potential slot-values, the algorithm then filters the set to obtain a consistent set of slot-value pairs. The first stage is to filter out any cases where the translated material overlaps. In these cases the more specific translation is retained, the others are dropped. Secondly there is a check to ensure that no slot is given the same value twice. If there is a conflict the higher weighted value is adopted.

It should be noted that the current algorithm does not check consistency of contextual constraints: different assumptions about the context may have been made during the filling of different slot values, and ideally the algorithm should check that each translation corresponds to a consistent path through the lattice. Despite this, the ability of the rules to deal with context and domain dependency, and to provide robust interpretation seem to give the system creditable performance.

7. Conclusions

This work points to various ways in which we can mix some of the advantages of linguistic analysis with shallower methods. The approach advocated incorporates linguistic information where necessary (e.g. for determining the scope of negation), but also allows linguistic constraints to be relaxed to ensure robustness.

Acknowledgements

This work was funded by the EU Project, Trindi (Task Oriented Instructional Dialogue, LE4-8314). I would like to thank other members of the Trindi Consortium for their comments on this work, in particular Robin Cooper, Ian Lewin, Manfred Pinkal and Massimo Poesio for various discussions which fed into the work. I would also like to thank Ian Lewin and Rob Koeling for their comments on an earlier draft.

References

Boye, J., M. Wiren, M. Rayner, I. Lewin, D. Carter and R. Becket, 1999 'Language-Processing Strategies and Mixed-Initiative Dialogues' in *Proceedings of IJCAI-99 Workshop on Knowledge and Reasoning in Practical Dialogue Systems.*

Bronneberg, W., H. Bunt, S. Landsbergen, R. Scha, W. Schoenmakers and E. van Utteren, 1980, 'The Question Answering System PHLIQA1', in L. Boye (ed.), *Natural Language Question Answering Systems*, Macmillan.

Carter, D., 1997 'The TreeBanker: a Tool for Supervised Training of Parsed Corpora' *ACL Workshop on Computational Environments for Grammar Development and Linguistic Engineering*, Madrid. Available as SRI Cambridge Technical Report CRC-068

Copestake, A, D. Flickinger, R. Malouf, S. Riehemann, and I.Sag, 1995, 'Translation using Minimal Recursion Semantics', in *Proceedings of the Sixth International Conference on Theoretical and Mehodological Issues in Machine Translation*, Leuven, Belgium.

Egg, M, J. Niehren, P. Ruhrberg and F. Xu, 1998, 'Constraints over Lambda-Structures in Semantic Underspecification', in Proceedings of COLING-ACL '98, Montreal, Canada, 353-359.

Goerz, G., J. Spilker, V. Strom, and H. Weber, 1999, 'Architectural Considerations for Conversational Systems -- The Verbmobil/INTARC Experience' to appear in the *Proceedings of First International Workshop on Human Computer Conversation*, Bellagio, Italy.

Hobbs, J., 1983, 'An Improper Treatment of Quantification in Ordinary English', in Proceedings of the 21st Annual Meeting of the ACL, Cambridge, Mass., 57-63.

Hobbs, J., S. Stickel, P. Martin and D. Edwards, 1988, 'Interpretation as Abduction', in *Proceedings of the 26th ACL*, Buffalo, New York.

Kasper, W., B. Kiefer, H.-U. Krieger, C. J. Rupp and K.L.Worm, 1999, 'Charting the Depths of Robust Speech Processing', to appear in Proceedings of the 37th ACL.

Lewin, I,, R. Becket, J. Boye, D. Carter, M. Rayner, and M. Wiren, 1999, 'Language processing for spoken dialogue systems: is shallow parsing enough?' in *Proceeding of Accessing information in Spoken Audio* (ESCA ETRW Workshop, Cambridge), April 1999. Available as SRI Cambridge Technical Report CRC-074.

Milward, D., 1991, *Axiomatic Grammar, Non-Constituent Coordination, and Incremental Interpretation*, PhD Thesis, University of Cambridge

Milward, D., 1994, 'Dynamic Dependency Grammar', *Linguistics and Philosophy, 17*, 56-1-605 van Noord, G., G. Bouma, R. Koeling and M-J. Nederhof, 1999, 'Robust Grammatical Analysis for Spoken Dialogue Systems' to appear in the *Journal of Natural Language Engineering*. Cmp-lg archive: cs.CL/9906026

Oaksford, M and M.Chater, 1991, 'Against Logicist Cognitive Science', *Mind and Language, 6,* 1-31.

Rayner, M., 1993, *Abductive Equivalential Translation and its application to Natural Language Database Understanding,* PhD Thesis, Royal Institute of Technology, Stockholm.

Reyle, U., 1993, 'Dealing with Ambiguities by Underspecification: Construction, Representation and Deduction', *Journal of Semantics*, 10:123-179.

Trujillo, A, 1995, 'Bi-Lexical Rules for Multi-Lexeme Translation in Lexicalist MT', in *Proceedings of the Sixth International Conference on Theoretical and Mehodological Issues in Machine Translation*, Leuven, Belgium, 48-66.

Whitelock, P., 1992, 'Shake-and-Bake Translation', in *Proceedings of COLING '92*, Nantes, France, 784-791.

Worm, K., 1999, 'A Model for Robust Processing of Spontaneous Speech by Integrating Viable Fragments', in *Proceedings of COLING-ACL '98,* Montreal, Canada, 1403-1407.

David Milward
SRI International
Suite 23, Millers Yard
Mill Lane
Cambridge
CB2 1RQ
G.B.
milward@cam.sri.com

Editing Speech Acts: A Practical Approach to Human-Machine Dialogue

Paul Piwek, Roger Evans & Richard Power

ITRI, University of Brighton

Abstract

This paper introduces a new approach to dialogue. This approach is based on the idea that dialogue can be seen as a form of two-person knowledge base editing. In a dialogue, it is the dialogue history which is being edited. We propose some constraints on editing dialogue histories which distinguish dialogue from knowledge base editing in general. We claim that our proposal yields both interesting conceptual insights into the nature of dialogue, and provides a starting point for building practically feasible dialogue systems. A first prototype of such a dialogue system has been developed. We provide a description of this system.

1 Introduction

In this paper we describe a new approach to dialogue. We demonstrate that this approach provides both a fresh view on the nature of dialogue and helps us to build practically feasible dialogue systems. Central to our approach is the notion of a shared dialogue history. The idea that utterances both give rise to updates of this history and are dependent on this history (e.g., for the interpretation of ambiguous expressions, pronouns, etc.) is well established (see, e.g., Clark, 1996).

Our main contribution is to take the idea that the utterances in a dialogue depend on and affect a shared history quite literally. A dialogue is modelled as a two agents editing a knowledge base (i.e., adding, cutting, copying and pasting knowledge objects) which represents the dialogue history. For this purpose, we use the WYSIWYM knowledge editing technology (see Power *et al.*, 1998). We show that a dialogue can be seen as a special type of two-person knowledge base editing. We provide a number of constraints on two-person knowledge base editing which distinguish a dialogue from other forms of knowledge editing, thus obtaining a new view on the nature of dialogue.

The approach that we describe has been developed in the context of the CLIME project.[1] The overall aim of this project is a generic web-based system for accessing legal information. We describe the application that is being developed within the CLIME project to illustrate the suitability of our dialogue model for web-based human-computer interaction.

We proceed as follows. In section 2, we describe WYSIWYM style knowledge base editing. Next, in section 3, we introduce our model of dialogue as knowledge base editing. In section 4, an application, which is an information system for maritime law, is introduced. We identify the sort of situations in which systems like the one we developed for the maritime domain could be useful. Section 5 contains a discussion of the relation between our approach to dialogue and the approach in current commercially available dialogue systems. Furthermore, at the end of this section we list avenues for further research.

2 WYSIWYM Knowledge Base Editing

Generally speaking, there are two ways of editing a knowledge base: by means of a command language and by means of direct manipulation. Let us first consider the use of command languages. A command language requires an interpreter which updates the knowledge base on the basis of the user's typed or spoken input. Ideally, the user should be able to edit the knowledge base by means of the language which s/he is most accustomed to, i.e., unrestricted natural language. Unfortunately, for the foreseeable future unrestricted text is not feasible for practical applications, because natural language understanding from free text is not yet sufficiently reliable. Therefore, so-called controlled languages have been introduced (see, e.g., Fuchs and Schwitter, 1996; van der Eijck, 1998). But there is a downside to the use of controlled languages: a user will have to learn to write in the controlled language, which can involve a substantial amount of effort.

Alternatively, there are direct manipulation interfaces to knowledge bases. Usually, the knowledge base consists of a network, which is graphically presented to the user (see, e.g., Paley, 1996). The user can now directly cut objects from the network and insert objects into the network. Thus each action by the user has a direct semantic interpretation. Hence, no parsing and interpreting is required. Again, there is also a downside. Network representations of knowledge can become very complicated and therefore difficult to understand by the user.

In Power *et al.* (1998), a new solution, called WYSIWYM (What You See is What You Meant), to the problem of knowledge editing has been proposed. WYSIWYM aims at the best of both worlds: the practical feasibility of direct manipulation and the ease of use of natural language text. The basic idea underlying WYSIWYM can be presented by means of a simple diagram, see Figure 1.

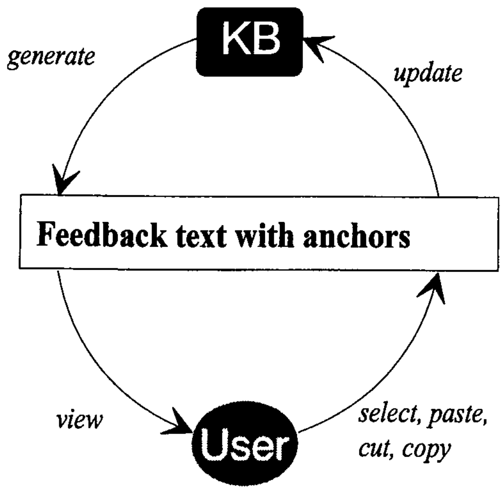


Figure 1: The editing cycle

Figure 1. represents the editing cycle. Given a Knowledge Base (KB), the system generates a description of the knowledge base in the form of a 'feedback text' containing 'anchors' representing places where the knowledge base can be extended. Each anchor is associated with pop-up menus, which present the possible extensions of the KB at that point. On the basis of the extension that the user selects, the knowledge base is updated and a new feedback text is generated from the new contents of the KB. Additionally, spans of feedback text representing an object in the KB can also be selected by means of the mouse. Cut and copy operations are available which allow the user to cut or copy the underlying knowledge base object into

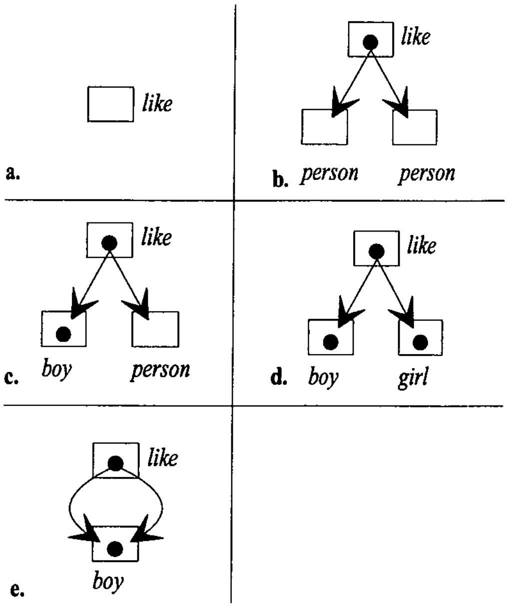


Figure 2: Editing a DAG

a buffer. Subsequently, such an object can be pasted into a location where the KB is still incomplete. After a cut or paste action, a new feedback text is generated which represents the updated KB.

Let us consider a very simple example which allows us to briefly illustrate the essential features of WYSIWYM editing. As usual, the knowledge base consists of two parts: a T(erminological)-box and an A(ssertion)-box. In the T-box, we specify the set of available concepts and their attributes:[2]

(1) a. `domain > [like, person].`
    b. `person > [man, woman, girl, boy].`
    c. `like intro [attr1:person, attr2:person].`

We start with the concept `domain`, which has two subconcepts: `like` and `person` (1.a). Subsequently, we introduce the subconcepts of the concept `person` in (1.b). Finally, in (1.c), it is stated that the concept `like` has two attributes. A concept following a colon indicates which objects are legitimate values of the attribute.

The A-box contains the actual knowledge that is to be edited. Formally, this knowledge has the structure of a Directed Acyclic Graph (DAG). The nodes in the graph stand for the instances of concepts, i.e., objects, and the directed arcs of the graph represent attributes. The basic editing operation on a DAG is that of adding a new object, of a specified type, as the value of an attribute of an existing object. However, when we start knowledge editing, the A-box may be empty. To start the editing process, we need a predefined root attribute in the A-box, which is not part of the T-box proper, but whose value has to be an instance of one of the concepts that have been introduced in the T-box. For instance, let us assume that value of the root attribute has to be an instance of the concept `like`. This means that the state of the KB before we start editing can be represented with the graph in Figure (2.a). On the basis of this KB a feedback text is generated:

(2) **Some relation of liking.**

The entire span of text is in boldface. This indicates that the text is an anchor. By clicking on it, a menu appears which shows the possible alternatives for expanding the KB. In this case, the menu has only one option: `new object`. When the user selects this option a new object of the concept `like` is introduced into the KB (see fig. 2.b). On the basis of the new KB a fresh feedback text is generated:

(3) **Some person** likes **some person**.

The new feedback text has two anchors. When the user selects the first anchor, a menu with the option `new object` appears again. But now, when user user clicks on `new object`, another menu appears, which allows the user to select the concept to which the object belongs:

man
woman
boy
girl

Let us assume that the user selects `boy`. The new KB (fig. 2.c) gives rise to the following feedback text:

(4) A boy likes **some person**.

Along the same lines, the second anchor can be expanded. This leads to a fully completed knowledge base (fig. 2.d) and the following feedback text:

(5) A boy likes a girl.

Alternatively, instead of inserting a new object into the incomplete network (fig. 2.c), the user could also have chosen to copy an object that was already available, and paste it into the part of the network that was not yet developed. For instance, the user could have selected the span of text 'a boy'. A menu would have appeared with the options `cut` and `copy`. If the user had selected `copy`, the underlying object would have been stored in a buffer. Subsequently, the user could have selected the span of text which represents the incomplete part of the KB, i.e., '**some person**'. A menu would have appeared with the option `paste`. By performing the paste action, the network in fig. (2.e). would have been constructed, and the following feedback text would have been generated:

(6) A boy likes himself.

In this case, a reflexive pronoun is generated for the second attribute which takes the boy in question as a value. Note that if `copy` and `paste` had simply operated on the graphemic level of the sentence (i.e., the character sequence) instead of the underlying semantics, the result would have been 'A boy likes a boy'. For a more detailed account of coreference and WYSIWYM editing we refer to van Deemter and Power (1998).

3 Dialogue and Knowledge Base Editing

In a dialogue, two persons produce utterances and thus build up a shared history. This history is grounded in the physical events that constitute the utterances. On a higher level of analysis, the history consists of the words that were used and the syntactic structure of the phrases in which the words occur. Let us call this the linguistic level. Finally, on the highest level of analysis, the history consists of a sequence of speech acts. The speech acts each have a speech act type or force and a semantic content (Searle, 1969).

In ordinary dialogue the interlocutors edit the dialogue history by producing new utterances which extend the dialogue history. They also edit the dialogue history at its higher levels of analysis: by producing new utterances they also produce new speech acts. In this paper, we want to examine what it would be like if the interlocutors could directly edit the dialogue history at its highest level of interpretation: the speech act level.

WYSIWYM allows us carry out this experiment. For that purpose, we assume that the KB represent the dialogue history at the speech act level. Unrestricted two-person editing of this KB would, however, be nothing like ordinary dialogue. We need three simple constraints on the editing process to arrive at something which is similar to ordinary dialogue. We will state these constraints and then discuss their implementation in a WYSIWYM environment.

Turn taking In a dialogue, people normally speak one at a time. In order to mimic this phenomenon, we need to prevent the interlocutors from simultaneously editing the dialogue history. Let us state this in terms of two simple constraints:

(C1) Interlocutors are prevented from editing the dialogue history simultaneously.

(C2) After an interlocutor has finished editing a speech act, s/he passes the turn to the other interlocutor.

For the moment, these constraint will do, although it is a simplification: in real dialogues overlapping speech does occur.

Immutability of history Once something has been said, the fact that it has been said is immutable.[3] In terms of knowledge editing this means that only the most recent speech act can be changed in the editing process.

(C3) Interlocutors are prevented from editing speech acts other than the most recent one.

Without this constraint in place we run into some serious difficulties. Suppose that it were allowed to edit speech acts further back in history. Note that some of the speech acts that occur after time some time *T* will depend on the speech acts before time *T*. This means that if a speech act before time *T* is edited, this also has repercussions for the ensuing speech acts. These repercussions can, however, not be overseen by one participant alone, since the ensuing speech acts were produced by both participants. For instance, if participant *A* were to change a question that preceded an answer produced by *B*, then this change would also affect the answer. Hence, the editing operation on the question would have no clearly delineated effect with respect to the entire dialogue history.

Note that although it is not possible to edit a dialogue history somewhere in the middle, this does not prevent us from going back to some point *T* in the history, and then pursuing a alternative history to the one that actually took place after *T*. This simply involves making a copy of the history from its starting point until *T*, and editing that copy. In section 4,we shall see that this can be useful feature for practical dialogue systems.

As a consequence of the constraints (C1), (C2) and (C3), interlocutors are prevented from editing one and the same speech act (whether it be simultaneously or at different times). Note again that this is a simplification: sometimes one interlocutor does complete another interlocutor's utterance.

Accessibility of history Although in ordinary dialogue, we cannot change previous utterances, we can make use of the content that they introduced, by means of, for instance, anaphora:

A: The pump is not working.
B: You have to replace *it*.

and presuppositions:

A: The pump is out of order.
B: Why *is it out of order*?[4]

In terms of knowledge editing this means that we are allowed to copy items from the preceding history and paste them into the speech act that is currently under construction. In this respect, knowledge editing seems to require no further constraint. Note, however, that as soon as the semantic representations associated with speech acts become sufficiently expressive further constraints are needed. We have in mind the constraints on accessibility of discourse referents (which correspond to objects in the knowledge base) as stated in Discourse Representation Theory (Kamp & Reyle, 1993). It is beyond the scope of this paper to address this issue, but see Kibble *et al.* (1999) and Power (1999).

Implementing Dialogue as Knowledge Editing Let us now describe how we have implemented these ideas by means of a WYSIWYM system. The idea is that the interlocutors are a user and a software agent (e.g., an expert system). The user and the software agent talk with each other via a *dialogue manager* which maintains the dialogue history. The dialogue history is represented as a DAG. In this DAG, there will be nodes representing speech acts. At the beginning of a dialogue we start with a DAG whose speech acts are not yet specified. In other words, the dialogue history still has to be filled in by the interlocutors. The interlocutors can, one at a time, expand a speech act, thus establishing step by step a dialogue history.

Consider the DAG in figure 3. This DAG represents part of a dialogue history which still has to be filled in. There are locations representing three speech acts: a query, an answer and a follow up query to the answer (in speech act terms, we have two directives and one representative). The dotted boxes indicate in what order the interlocutors are allowed to edit the incomplete dialogue history. We start at the innermost box. Technically, this means that the root attribute is set to the location in the innermost box.

Let us assume that the user takes the first turn. The effect of having the `query` location as the value of the root is that the WYSIWYM generator will only generate a feedback text for this node. The feedback text is:

User: **Some enquiry.**

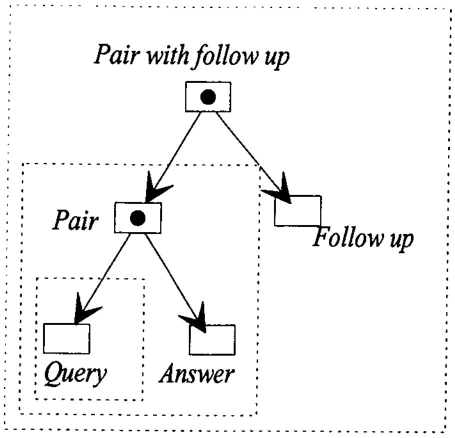


Figure 3: Incomplete DAG representing a dialogue history

The user can now start editing this query, until s/he is satisfied with the result, e.g.:

User: I have a tanker with an opening in the upper deck. The opening is elliptical in shape (400x600mm). Is this acceptable?

Now that the user has finished editing the query, s/he can hand over the turn to the other interlocutor. This means that the user interface will have a button for this purpose. The software agent then constructs a representation of its answer (of course, it doesn't use the WYSIWYM interface, since the agent has been build in such a way that it can only construct well-formed representations). The software agent submits this answer to the dialogue manager, who slots it into the appropriate location of the dialogue history (the `answer` location).

Now, it is the user's turn again. At this point the root attribute is set to the `pair with follow up` location. Note that here the root is not set to an incomplete location, but rather to an object. The effect of this is that the generator will only generated the feedback text for the knowledge objects below this object. Thus a feedback text is generated which represents the full query and its answer, and an editable span of text for the follow-up query:

User: I have a tanker with an opening in the upper deck. The opening is elliptical in shape (400x600mm). Is this acceptable?

System: Yes it is acceptable without reinforcements.

User: **Some follow up question.**

At this point, the user can edit the `follow up` location and copy objects which are situated under the `pair` node into the follow up location. In other words, although the content of the preceding speech acts can no longer be edited, it is available for constructing the new speech act. For instance, the user might expand the follow-up location into a why question:

User:	I have a tanker with an opening in the upper deck. The opening is elliptical in shape (400x600mm). Is this acceptable?
System:	Yes it is acceptable without reinforcements.
User:	Why is **something** the case?

At this point, the user can copy the proposition 'it is acceptable without reinforcement' into the location '**something**'. Thus, we end up with the following feedback text:

User:	I have a tanker with an opening in the upper deck. The opening is elliptical in shape (400x600mm). Is this acceptable?
System:	Yes it is acceptable without reinforcements.
User:	Why is it acceptable without reinforcements?

For the purpose of this example, we have identified the incomplete location of the dialogue history with very specific speech acts (query, answer, follow-up). We could also have chosen to use much broader types of speech acts as locations. For instance, we could have simply identified a location with a speech act of any type, and then let the user, by means of the WYSIWYM menu, choose whether it should be a question, conclusion, promise, declaration, etc. For a practical application, where the sequence of speech act types can be predicted reliably in advance, such freedom would, however, only have a negative effect on the ease of use of the system. Furthermore, if we were to use more a more general taxonomy, we would be faced with the problem of selecting an appropriate one. Traum (1999) points out that one needs to consider the task at hand when selecting a speech act taxonomy. For our purposes, a taxonomy which closely follows the intuitions of ordinary language users is most desirable.

There is one limitation to the system as we have described it here. The range of possible speech act *types* that can be constructed at a given point in the dialogue does not depend on the preceding dialogue history (note, however, that the *content* of a speech act can depend on what has been said before, since the user is allowed to copy material from the preceding dialogue). This range is determined by the networks that the T-Box allows us to build. This T-box is static during the dialogue. We might, however, want to allow the user to only ask a follow-up question of a particular type, after an answer of a particular type. This can be achieved by giving the dialogue manager a more active role. In addition to simply moving the root attribute to the next incomplete speech act location, the dialogue manager might pre-edit this speech act on the basis of the information it has concerning the preceding dialogue, or simply hide certain, no longer relevant, ways to expand the location from the user.

For instance, suppose the system has just replied 'No' to the question 'Are there any pumps on this ship?'. In that case, a where-question ('Where are the pumps located') is no longer a sensible follow-up. Alternatively, if the answer had been 'Yes', a where-question would have been a natural follow up. Here it is useful if the system can dynamically decide whether to allow the user to construct a where-question or not. This idea has, however, not yet been implemented in our first prototype.

4 An Application

The application that is described in this section is being developed in the context of the CLIME project whose aim is to develop software to support access to legal and regulatory information. The concrete application that is under development is in the area of maritime law.

Let us first describe the context of use of the application. The potential end users of the application are the surveyors of a classification society. The business of this society is the production and application of rules, in particular, in the maritime domain. In order for a ship to be insured, the owner of the ship has to make sure that the ship has been certified by at least one such classification society. Certification is carried out by the surveyors, who check whether a ship fulfils all the rules that are laid down by the classification society.

Currently, the rules of the classification society are available both in paper form and on CD-Rom (with simple information retrieval functionality). Our project partner at the university of Amsterdam has developed a legal reasoning system which can apply (a formalisation of) the rules to descriptions of ships (see Winkels *et al.*, 1998). The system can, given a (partial) description of a ship, determine which rules apply to it, and whether the described situation is acceptable according to the rules or not.

Requirements On the basis of discussions with the end-users of the system (i.e., experts of the French classification society Bureau Veritas) we established a number of *requirements* to which we have geared the architecture of our application prototype:[5]

- The system aims to help the surveyors in their daily practice of certifying ships. This means that the system should be available on site, for instance, during an inspection at a ship yard. Therefore, a web-based architecture has been chosen for the system. The user down loads the user interface as a JAVA applet into his/her computer, whereas the system itself is running on a possibly more powerful server elsewhere (legal information serving is computationally expensive for real life domains).
- The system is used in an internationally operating company and should be adaptable to the language of the local users.
- The user can ask queries and follow-ups to his/her queries, even days later than the original query.
- Given the nature of the domain, i.e., legal/regulatory information, there is a preference for having access to information in its written form.

The Architecture A simplified representation of the system architecture is depicted in Figure 4. The legal information server and the dialogue database are located on a server. The user interface to the dialogue manager is running on the user's computer as a client, whereas the main functionality provided by the dialogue manager is running on the server. The idea is that 1. the user constructs a query using WYSIWYM. 2. This query is stored in the dialogue database. 3. The query is submitted to the legal information server. 4. The legal information server returns back an answer, which is integrated into the dialogue history (up till that point it only contained the user's query) which the dialogue manager retrieves from the dialogue database. 5. Subsequently, the dialogue manager stores the updated dialogue history in the

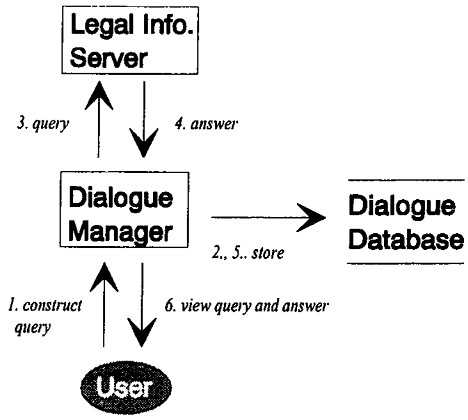


Figure 4: The system architecture

dialogue database. 6. The user can view (the feedback text corresponding to) the new dialogue history (the user is notified by the system of any changes to the dialogue database), and if needed continue to edit it.

Database-oriented dialogue There are two reasons for maintaining a dialogue database. Firstly, it allows the user to quickly construct queries which deviate not too much from queries which s/he submitted earlier on. For that purpose, the user simply retrieves a query from the database which is sufficiently similar to the one s/he wants to submits and modifies this query before submitting it. The second reason for maintaining a dialogue database is that this makes the dialogue independent of a persistent internet connection between the user's computer and the server. It is now possible to construct a follow up query to a query which was posted hours, days or even weeks ago. Furthermore, the legal information server may sometimes take more than an hour to compute an answer. It would be rather awkward if the user needed to remain logged in to wait for such an answer. Fortunately, this is not necessary because his or her query is stored in the dialogue database and updated with the answer whenever the legal information server has found one. The next time the user logs into the system s/he is notified of the change in the dialogue database (alternatively, the user may also be notified immediately by email) and can inspect the query and its answer.

Example of a session with the system After a user has logged in to the system, a main window appears. This window (see figure 5.) is divided into two panels: a panel with a directory structure and a view panel. The directory structure consists of folders and files in which dialogue histories are stored. By clicking on a file, the user can view the dialogue history which it contains in the view panel. Thus it is possible for the user to browse through the collection of dialogue histories which s/he has built up by querying the system.

On the menu bar, there is a file, query and options menu. On the file menu, the user finds the usual operations for manipulating files and folders, such as 'delete', 'rename, open, close folder, etc. The query menu provides amongst other things the possibility to create new dialogue histories, edit existing dialogue histories and extend existing dialogue histories (i.e., construct follow-ups). For instance, when the user selects the option *edit*, a WYSIWYM editing window pops up. Within this window the user can then make alterations to the dialogue history which was depicted in the view panel (see figure 6.). When the user is satisfied with the changes which s/he made and has constructed a complete query (i.e., the last query in the

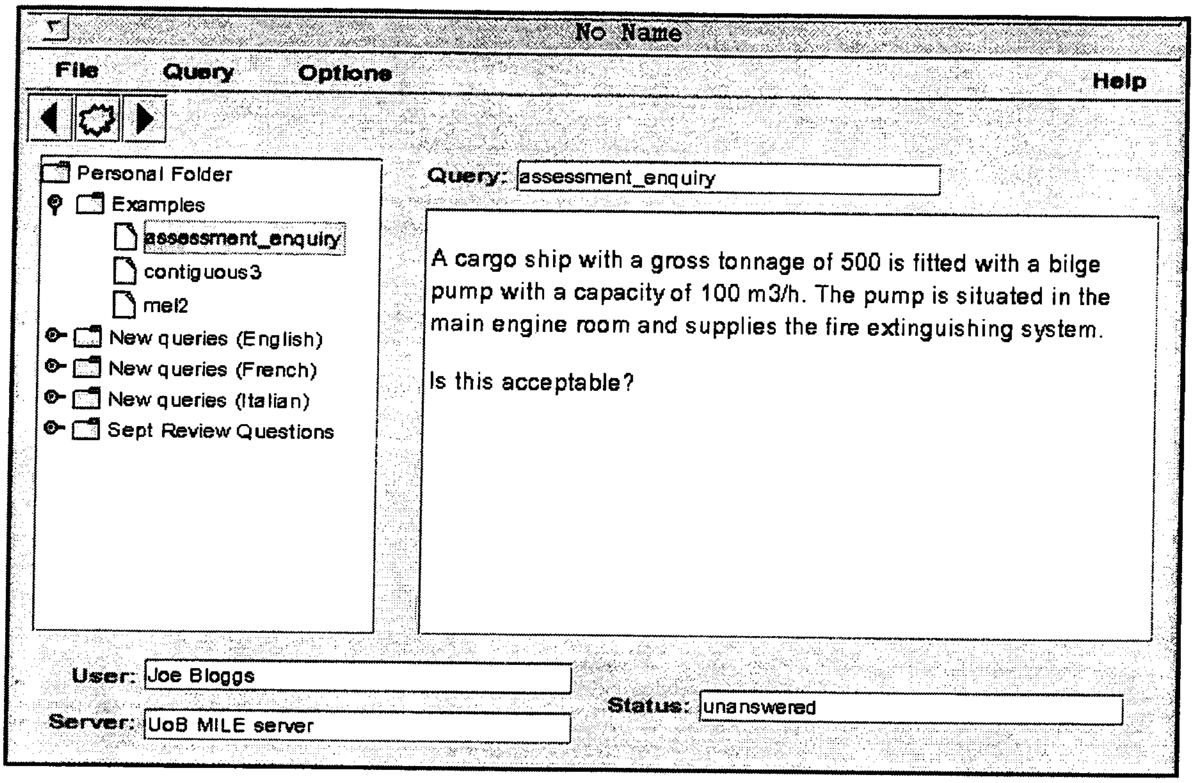


Figure 5: The main window

dialogue history contains no locations which still need to be filled in), s/he can send the query to the legal information server by selecting submit on the file menu of the editing window.

Let us now have a closer look at the query in figure 6. The current system has a T-Box and a grammar which covers one chapter of the rules book of the classification society (there are 30 chapters in total). The example is modelled after the following query which was provided by experts from the classification society: 'I have a cargo ship with a gross tonnage of 500, that is fitted with a bilge pump with a capacity of 100m3/h in the main engine room. This pump is driven by the main engine and can provide water to the fire extinguishing system. Is this acceptable?'. Currently, a user can construct this query with the WYSIWYM system. The feedback text that is generated by the system is depicted in figure 6. (The feedback text is not yet fully complete. Note that the phrase 'some pump' is printed in a lighter grey. These are anchors which indicate in which respects the underlying content of the query is still incomplete).

On the basis of a formal representation of the answer to this query (the formal representation was handmade, because the legal information server and the dialogue manager have not yet been integrated) the text in Figure 7. is generated.

It is beyond the scope of this paper to go into the formal details concerning the representations of answers (but see Winkels *et al.*, 1998). Let us, however, briefly sketch how the surface structure of the answer is related to its underlying formal representation. The underlying representation makes reference to two rules which are applicable to the case that the user wants to examine. The first rule says that the firepumps of a ship are not allowed to be driven by the main engine. A second rule says that it is allowed that a firepump is driven by the main engine, if the gross tonnage of the ship is less than a 1000. Thus we have (where C = it is allowed):

$P \rightarrow \text{not } C$

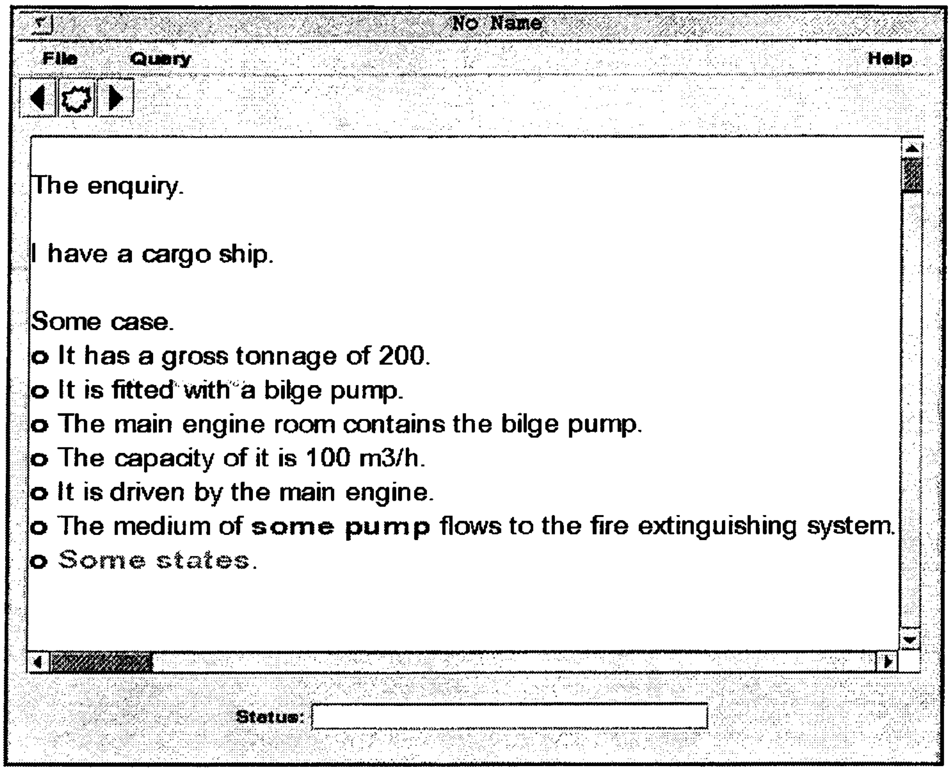


Figure 6: The dialogue history editing window (English)

$(P \wedge Q) \rightarrow C$

The rule with the more restricted antecedent wins. We have used the following template to associate this type of answer with a natural language text (where $l(P)$ stands for the translation into natural language of the proposition P):

Although $l(P)$, $l(C)$, because $l(Q)$.

When the legal information system comes back with the answer, it is fitted into the representation of the dialogue history. The system then generates a text from the representation of the antecedent dialogue history which has been merged with the answer. It is essential that the antecedent history and the answer are merged in order to get cross speaker anaphora right. For instance, in figure 7., 'the bilgepump' is used to refer to an object which was introduced in the user's query. In the underlying representation, the bilgepump which is referred to in both the query and the answer is represented with one object (i.e., a discourse referent).

In Figure 7., we also see an example of a follow-up query. This follow up query has been constructed starting from the following feedback text: 'Can you tell me why **some answer state**?'. The user can fill the location represented by the span '**some answer state**', by selecting a part of the answer, copying the underlying knowledge object, and pasting it into the follow up. Thus a complete follow up question is constructed.

Finally, let us note that because the feedback texts which represent the dialogue history are generated from an underlying language independent representation, it is rather straightforward to change the language which is supported by the system. All that needs to be done, is to change the natural language generator which produces feedback texts from the formal representation of the dialogue history. The user can select a language on the options menu of the main window. For instance, if the user at this point selects French, the text in the query window is replaced with its French equivalent (see figure 8).[6]

The answer.

Although the bilge pump is a firepump and it is driven by the main engine (see 20–052–11), the case is allowed, because the cargo ship has a gross tonnage of less than 1000 (see 20–052–31).

A follow up.

Can you tell me why the bilge pump is a firepump?

Figure 7: Answer and a follow-up

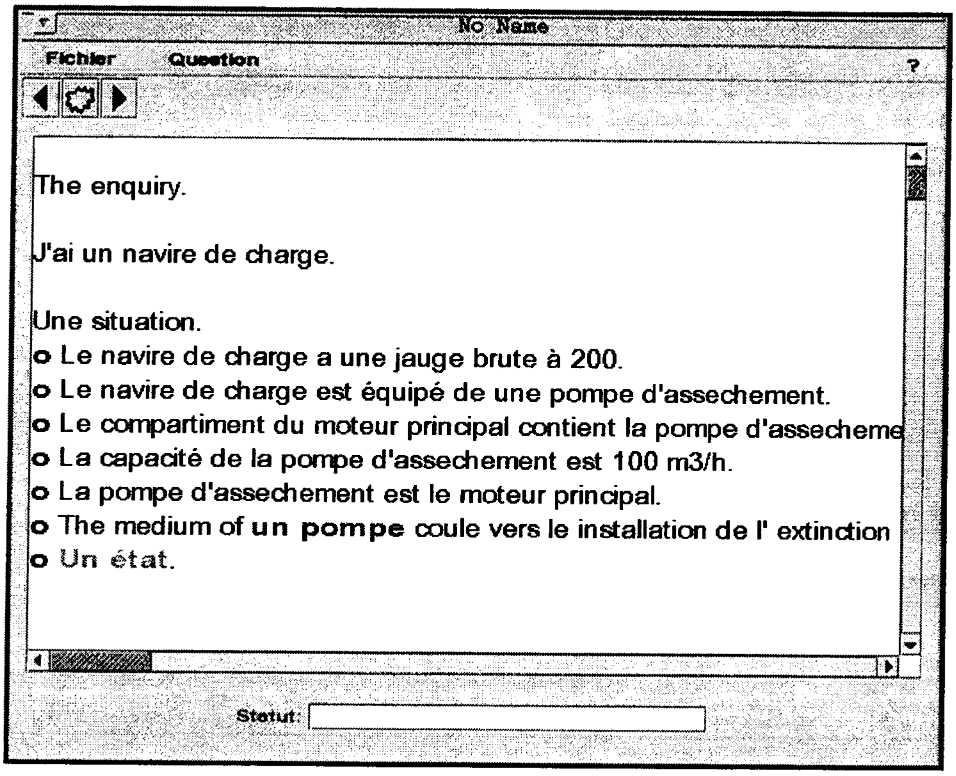


Figure 8: The dialogue history editing window (French)

5 Discussion and Conclusions

In summary, the model of dialogue as knowledge editing that we have introduced provides a fresh view on dialogue. In particular, by exploring the differences between dialogue and knowledge editing in general, we obtain a new perspective on the nature of dialogues. Furthermore, the model is promising for use in practical applications. It is based on the user friendly interaction provided by WYSIWYM editing. In combination with the use of a dialogue database, it has some features which make it particularly useful for human-machine dialogue over the internet. In the remainder of this section we compare our approach with alternative approaches to natural language interpretation in dialogue, and conclude with some remarks on directions for further research.

Natural language interpretation is a notoriously difficult task. Textbooks on human-computer interaction (e.g., Sutcliffe, 1995; Dix et al., 1998) often point out that it seems unlikely that a general natural language interface will be available for some time, if at all. This claim is based on the fact that natural language interpretation is heavily context-dependent. In particular, interpretation may require any kind of general background knowledge that is available to a human agent. Formalization and tractable implementation of such immense knowledge resources seems implausible for the foreseeable future (currently, the only large scale project aiming at that goal is the Cyc project, see Guha & Lenat, 1994).

However, natural language interpretation seems feasible for small scale domains. In recent times, commercially viable dialogue systems have been developed for such constrained domains. The Philips automatic train timetable information system (Aust *et al.*, 1995) is a case in point. However, the success of this and similar systems depends on the fact that there is a very limited number of simple informational items which the systems needs to elicit from the user in order to satisfy the user's request. For instance, in order to provide the user with a train connection, the Philips system needs the place of departure, the destination and the time at which the user wants to depart or arrive. The system can provide a connection when it has filled the slots for these parameters.

Suppose that the user tells the system "I'd like to go to Hamburg". In that case, the only part of the utterance that the system actually understands is "to Hamburg". It uses this information to fill in the slot for the destination. Thus, the system could not have told the difference between the aforementioned utterance and "I'd not like to go to Hamburg". Furthermore, the Philips system does not deal with anaphoric references such as pronouns and descriptions. These limitations make this type of approach unsuitable for the CLIME domain.

In CLIME, the information that the system needs to elicit from the user typically consists of multiple informational items which may be linked by co-reference and enter into logical (e.g., negation) and discourse relations (e.g., the relation between a legal case and a question about that particular case). Consider, the following query in the maritime domain which illustrates all three of the aforemention relations:

> I have an oil tanker. One of its bilge pumps is not functioning. Can I replace the pump with an ejector system?

The technology that is used in current state of the art applied dialogue systems is not suited for this type input. A legal analysis system will have to understand the meaning of the user's utterance on a much more detailed level than the aforementioned dialogue systems do in order to provide an answer to the user's query. In particular, the system has to recognize coreferential terms and the logical and rhetorical structure of the query. We already pointed out that for the foreseeable future unrestricted text seems to be not feasible for practical applications.

We hope to have made it plausible that our approach to dialogue as knowledge editing, and in particular, the use of the WYSIWYM technology, provides a promising alternative to the approaches taken in currently available commercial dialogue systems and research dialogue systems (the latter often rely on full parsing and interpretation of a user's utterances, but see Milward, 1999). Note that there is an interesting connection between WYSIWYM editing and the frame-driven or form filling approaches which are characteristic of commercially available dialogue systems (The frame-driven approach goes back to Bobrow *et al.* (1971). Its recent success seems to be partly due to the advent of high quality speech recognition technology). When a user constructs a DAG by means of WYSIWYM, the user can be seen as filling in slots. However, in doing so, the user builds up a DAG whose complexity goes beyond that of a simple frame structure. Firstly, the DAG may be build up using recursion (in other words, when we construct a DAG, we may fill in an indefinite number of slots, whereas frames will always have a fixed number of slots), and secondly we can copy parts of the dag and insert them into other locations in the DAG which have not yet been filled in (we make essential use of this feature to model coreference).[7]

Whereas building up a structure using recursion and copy and paste can be done by means of direct manipulation, these features seem to make it difficult to see how DAG's could be successfully used in spoken natural language dialogue systems, without having to fall back to full parsing and interpretation of the user's utterances.

Finally, let us indicate some issues which we would like to address in the future:

- Currently, a demonstrator of the dialogue manager exists. It has not yet been integrated with the full application. However, the first trials with end-users of the system have been very encouraging. Three end users were asked to formulate queries using the WYSIWYM technology and had no difficulties in doing so. Within the coming months, a formal evaluation of the system has been planned.

- We would like to further explore the possibilities of making the range of utterances that the user can construct at a given point in the dialogue more dependent on the antecedent dialogue history.

- The issue of sub dialogues has not been addressed in this paper. In our application domain these can, however, not be avoided (e.g., sometimes the legal information system needs to obtain more background information from the user before it can answer a question).

- Finally, we have been working on extending the expressive power of the formal language which is used to represent dialogue histories. Power (1999) shows how we can extend it to include Discourse Representation Structures (Kamp & Reyle, 1999), and recently a prototype which can deal with reference to plural objects has been devised.

Acknowledgements

We would like to thank Kees van Deemter for commenting on an earlier version of this paper. We would also like to thank the audiences at Amstelogue '99, the ITRI CL Meeting (University of Brighton) and the IMS Colloquium (University of Stuttgart) for useful comments and discussions. The front-end JAVA interface which is depicted in section 4 was implemented by Neil Tipper with help from Chris Douce.

PAUL PIWEK, ROGER EVANS AND RICHARD POWER

Information Technology Research Institute (ITRI)
University of Brighton, Lewes Road,
Brighton BN2 4GJ
UK
e-mail: firstname.lastname@itri.brighton.ac.uk

Notes

1. CLIME stands for Computerized Legal Information Management and Explanation. It is funded by the EC Esprit Programme under project number EP 25.414. The participants in CLIME are British Maritime Technology Ltd., Bureau Veritas, TXT Ingegneria, The University of Amsterdam and the University of Brighton.

2. We use the ProFit notation. See Erbach (1995).
3. Note that this does not mean that a participant can not contradict something that s/he said earlier on, but merely that the fact of having said it can not be 'undone'.
4. The question 'Why P?' is said to presuppose that P.
5. This list contains only a selection of the requirements. The full list of requirements can be found in Piwek *et al.* (1999:6).
6. Currently, our lexicon for French is not yet complete. Whereever the generator cannot find the appropriate French word, it defaults to English.
7. It would be fairly straightforward to devise, for instance, a WYSIWYM system for the train information domain. In the T-box we would need a concept `journey` with attributes whose values are a place of departure, a destination and departure time. On the basis of an object belonging to the concept journey, the system could then generate a feedback text saying 'I want to travel from **some place** to **some other place** at **some time**'. The user would subsequently have to expand the anchors which are in boldface. The final text might then be: 'I want to travel from Brighton to Gatwick on the second of August at 10:00 am.'

References

Aust, H., M. Oerder, F. Seide and V. Steinbiss (1995), 'The Philips automatics train timetable information system', Speech Communication, 17, 249–262.

Bobrow, D., R. Kaplan, M. Kay, D. Norman, H. Thompson and T. Winograd (1971), 'GUS, A Frame-Driven Dialog System', Artificial Intelligence, 8:2, 155–173.

Clark, H. (1996), *Using Language*, Cambridge University Press, Cambridge.

Dix, A., J. Finlay, G. Abowd, R. Beale (1998), *Human-Computer Interaction*, Prentice-Hall, Hertfordshire.

Erbach, G. (1995) 'Prolog with features, inheritance and templates', *Proceedings of EACL-95*, Dublin. 1995, 180–197.

Fuchs, N. and R. Schwitter (1996), 'Attempto Controlled Language (ACE)', *First International Workshop on Controlled Language Applications*, Leuven, Belgium.

Guha, R. and D. Lenat (1994), 'Cyc: A Mid-Term Report', AI Magazine, 11(3), 32-59.

Kibble, R., R. Power and K. van Deemter (1999), 'Editing logically complex discourse meanings', *Proceedings of the 3rd International Workshop on Computational Semantics*, 147–162.

Milward, D. (1999), 'Towards a Robust Semantics for Dialogue using Flat Structures', *Preproceedings of Amstelogue '99.*

Paley, S. (1996) 'Generic Knowledge-Base Editor User Manual', Technical Report, SRI International, California.

Piwek, P., N. Tipper & R. Evans (1999), 'Specification of the Natural Language Software Modules', CLIME Deliverable D.1.4, University of Brighton.

Power, R., D. Scott and R. Evans (1998), 'What You See Is What You Meant: direct knowledge editing with natural language feedback', *Proceedings of ECAI-98*, Brighton, UK, 1998, 180–197.

Power, R. (1999), 'Controlling logical scope in text generation' *Proceedings of the European Workshop on Natural Language Generation*, Toulouse, France.

Searle, J. (1969), *Speech Acts: An Essay in the Philosophy of Language*, Cambridge University Press, Cambridge.

Sutcliffe, A.G. (1995), *Human-Computer Interface Design*, Second Edition, Macmillan, London.

Traum, D. (1999), 'Extended Abstract: 20 Questions on Dialogue Act Taxonomies', *Preproceedings of Amstelogue '99.*

van der Eijck, P. (1998), 'Controlled Languages in Technical Documentation', *Selected Papers from the Eight CLIN meeting*, Rodopi, Amsterdam, 187–203.

van Deemter, K. and R. Power (1998), 'Coreference in knowledge editing', *Proceedings of the COLING-ACL workshop on Computational Treatment of Nominals*, Montreal, Canada, 1998, 56–60.

Winkels, R., A. Boer, J. Breuker & D. Bosscher (1998), 'Assessment Based Legal Information Serving and Co-operative Dialogue in CLIME' *Proceedings of JURIX-98*, GNI, Nijmegen, Netherlands 131–146.

Annotating Conversations for Information State Updates

Massimo Poesio[†], Robin Cooper[‡], Staffan Larsson[‡],
Colin Matheson[†], and David Traum[♠]
[†] University of Edinburgh, [‡] Göteborg University, [♠] University of Maryland

Abstract

We present an experiment in annotating a dialogue using two different notions of information state: a stripped down variant of Ginzburg's view of the dialogue game board, including questions under discussion (QUD), and the dialogue model of Poesio and Traum which builds upon previous work by Traum on the grounding process. In this first experiment we annotated a task oriented dialogue with the two notions of information state. We also present two tools useful for this kind of annotation: an annotation tool called *TranScript*, and the *Thistle Diagram Editor*. While annotating for information states is not suitable for large-scale annotation, we do feel that the methodology we are developing could be useful both for people who are interested in studying dialogue acts and for people who are building dialogue systems.

1 Introduction

Probably the central issues in analysis of dialogues are the questions of *why language participants say what they say*, and *what are the effects of those utterances.* These questions are obviously closely linked, because much of the reason for saying something is based on what has been said before. Modeling at least some aspect of the answers to these questions is also crucial for designing computational systems to engage in dialogue: these systems need to have procedures for determining what to say next and how to update their internal state on the basis of utterances. The general problem of coping with these issues is often termed *dialogue management*, and the components of the system most centrally concerned with these questions are termed *dialogue managers.* There are many different ways to model the process of finding these answers, ranging in degrees of complexity and closeness of approximation to human processing.

One very simple strategy is to either just produce particular utterances in sequence, or directly compute a response on the basis of the preceding utterance from the user. This is the strategy adopted by Eliza and other very simple programs. The problem with this approach is that often a context of more than just the previous utterance is needed to produce an appropriate next utterance. A more sophisticated approach involves using a *grammar* of acceptable dialogues, usually encoded as a finite state or recursive transition network, where the utterances represent transitions between states, and the states represent the context needed to decide what to say next. This approach also has its limitations, since it may lead to very large networks if all of the necessary context is encoded by differences in states. One common approach is to treat the utterances as encoding one of a limited set of abstract *moves*, and transitions are specified in terms of these moves, with other information being represented in other ways (e.g., variable or data structure values).

A more general approach is to view the dialogues in terms of the relevant *information* that the dialogue participants have (perhaps in addition to a notion of state in a network). From this vantage point, the main effect of an utterance is to change this information in some way, and the information is used by the participants to decide what to do next. The big question, then, is what kind of information is useful for this process. We can classify dialogue related information into two broad categories: *static*, which contains information critical to behaving appropriately in the dialogue, but does not actually change in the dialogue itself, and *dynamic*, which does change as the dialogue progresses, often after each utterance or sometimes in between utterances. The static information state will include both information about the domain such as how to do things, as well as meta-information about the dialogue participation process, such as how to update the dynamic state. The dynamic state will include the actual changes that motivate particular actions. *Moves* can now be seen as optional, since, while they might compactly serve to indicate the set of updates to the information state, one could also more directly represent the information change coming from the utterance without classifying the latter into one move or another.

The information state approach can also easily model the previous approaches, as well. For example, the connectivity of the network, would be the static information state, while the dynamic information state would be the particular current state, as well as any other information that might be useful for determining a next transition.

There are also different approaches to modeling the information state in terms closer to the dialogue itself or the mental and interactional states of the participants. In terms of mental states of dialogue participants, common mental attitudes include *Belief* (the participants' model of the world), *Desire* (what the participant wants the world to be like), and *Intentions* (plans the participant has developed for how to change the world). These are also often augmented with other attitudes, such as *Options* (ways that the agent can change things), and social states, with more than one agent involved. These latter include "mutual belief" or *common ground* which represents information that the participants believe to be shared, as well as various sorts of social commitment of one agent to others, including actions to be performed, or representations of how things are.

Other types of information refer more to the situation of the dialogue itself rather than the mental states of the participants. These include the *turn*, or which participant has the right to speak, and some notion of the *topic structure*, or what the participants are currently talking about. This notion is conceived of in many different ways, for example, in terms of the *intentional structure* [14] of how current topics relate to overall objectives, or in terms of the questions under discussion [12], which licenses what kinds of utterances may be made and understood.

A major goal of the TRINDI project is to be able to precisely characterize information states in dialogue, as well as their relationship to moves and providing answers to the important questions mentioned above. Doing this may provide a sound basis for empirical studies on which sorts of information states may be necessary or sufficient for engaging in particular kinds of dialogues.

2 Characterizing Information States

The characterization of the state of the conversation adopted in the spoken dialogue systems currently in real use, or close to actual use (e.g., [1]) can be represented in terms of feature structures as in (1): a list of fields which the system must fill before being able to ask a query.

(1) $$\begin{bmatrix} l_1 & = & a_1 \\ l_2 & = & a_2 \\ \ldots & & \\ l_n & = & a_n \end{bmatrix}$$

For example, in the Autoroute domain, the goal of the system is to identify the start and end points of the trip, and the departure time; this information can be represented as in (2).[1]

(2) $$\begin{bmatrix} \text{START} & = & \\ \text{END} & = & \\ \text{STIME} & = & \end{bmatrix}$$

This notation can be interpreted in various ways. One interpretation we have adopted is that in terms of typed records as discussed in [6, 5]. Using the notation $a : T$ to represent the judgment that a is of type T, if $a_1 : T_1, a_2 : T_2, \ldots, a_n : T_n$ then the object in (1) is of the record type in (3).

(3) $$\begin{bmatrix} l_1 & : & T_1 \\ l_2 & : & T_2 \\ \ldots & & \\ l_n & : & T_n \end{bmatrix}$$

Updates to these information states can be formalized as operations on these features structures, which can be simply setting of values for the fields in the simple example in (2). Feature values are also allowed to be more complex types, including stacks, lists, or other records. In this more complex case, updating the information state amounts to performing the appropriate update operation for the specified field.

In addition to task-specific aspects of the information state, such as that expressed in (2), it is very important to represent the state of the participants themselves, which is needed to interpret and participate coherently in a dialogue. There are several different dimensions to this state, which can be conveniently represented as hierarchical records and fields.

A main concern is whose information state is being represented. For dialogues with two participants, **A** and **B** there are three options: **A**'s state, **B**'s state, and an external "objective" state. When things are running smoothly, these will all tend to converge, however they may diverge in cases of un-repaired misunderstanding. Even when things are going well, there will be short-term differences in the information state, e.g., when **A** has decided what she will say but before she has said it. We take a middle ground between these three perspectives, representing an "objective hypothesis" of the information state of each participant, though not representing the participants views of the information state of the other participant. Thus, for the two-party dialogues we will be annotating, the top-level information state of the dialogue is a record with two fields, one for the information state of each participant.

Within each agent, there is also the question of how that agent views the commonality of the information: whether it is information private to the speaker, or shared between the participants. There may also be "quasi-shared" information which is accessible to all in some way, but not demonstrated or perhaps even assumed to be shared (yet).

Within each modality, there are also the individual types of information, themselves, represented variously as sets, lists, etc. Thus the kinds of information states we are looking at are generally records of the following structure:

$$(4) \quad \begin{bmatrix} \text{PARTICIPANT A} & : & \begin{bmatrix} \text{modality}_1 & : & \begin{bmatrix} \text{infotype}_1 & : & T_1 \\ \dots & & \\ \text{infotype}_k & : & T_k \end{bmatrix} \\ \dots & & \\ \text{modality}_n & : & \begin{bmatrix} \dots \end{bmatrix} \end{bmatrix} \\ \text{PARTICIPANT B} & : & \begin{bmatrix} \dots \end{bmatrix} \end{bmatrix}$$

Updates of individual aspects of the information state can be represented using the appropriate update operation and record location. E.g., for an information state of the type in (4), assuming T_k is the type *stack*, then (5) would be an example update operation:

$$(5) \quad \text{popRec}(\text{PARTICIPANT A.modality}_1\text{.infotype}_k)$$

More complex updates can be handled with sequences of such operations.

3 Two examples of annotation schemes

In this section we discuss in more detail how we have used this methodology to develop annotation schemes to study two taxonomies for classifying dialogue acts, one developed by Cooper and Larsson on the basis of work by Ginzburg [7, 13], and one developed by Poesio and Traum [18]. The discussion uses example annotations of a dialogue from the Autoroute corpus, listed in Appendix A.

3.1 Scheme 1: The Cooper-Larsson model of Information States

In this section we present a model of information states, using a stripped down variant of Ginzburg's [10, 11, 13] view of the dialogue game board, including questions under discussion (QUD). The development strategy has been to start as simply as possible and to add additional complexities only as they are required for representing the features of the dialogues in question. In particular, the instantiation of (4) for this information state type is shown in (6).

$$
(6) \quad \begin{bmatrix} \text{PRIVATE} & : & \begin{bmatrix} \text{BEL} & : & \textit{Set(Prop)} \\ \text{AGENDA} & : & \textit{Stack(Action)} \end{bmatrix} \\ \text{SHARED} & : & \begin{bmatrix} \text{BEL} & : & \textit{Set(Prop)} \\ \text{QUD} & : & \textit{Stack(Question)} \end{bmatrix} \end{bmatrix}
$$

That is, we made a division between PRIVATE and SHARED information. The private information consisted of a set of private beliefs (a set of propositions). Propositions are represented as English sentences with deictics referring to the dialogue participants replaced by the labels A and B. At the level of detail we were aiming at in this analysis it did not seem relevant to commit to one particular formal semantic theory. We are more interested in the dynamic modifications to the various fields in the information state rather than the exact formal representation of the objects.

The second private field is an AGENDA which is a stack of actions which the agent is to perform. The idea here is that the agenda represents very local actions. More general goals that the agent wishes to achieve with the conversation (or her life) would, on the simple view presented here, be included in the private beliefs[2]. In contrast to goals, agenda items are actions that should in general be performed in the next move. Agenda items are introduced as a result of the previous move.

The first SHARED field in the information state is again for a set of beliefs (i.e. a set of propositions). It is something of a misnomer to call this beliefs since it is meant to represent what has been established for the sake of the conversation and we do not really mean that this necessarily represents a commitment on the part of the dialogue participants to the shared propositions. The shared beliefs represent rather what has been established as part of the conversational record, assumptions according to which the rest of the dialogue should proceed. This can, of course, be distinct from what the dialogue participants "really think".

The second SHARED field is QUD, a stack of questions under discussion. Like the agenda, this is meant to be a local affair, representing question(s) that should be addressed more or less in the next turn and not general issues that have been raised by the conversation so far or issues that the agent feels to be generally relevant.

We tried to make minimal assumptions about what actions could be put on the agenda (i.e. what actions could be performed by the dialogue participants). We characterize possible actions informally by the following inference rules, assuming that we have a type *Question* and a type *Prop(osition)*.

(7) $\dfrac{q\text{:Question}}{\textbf{respond}(q)\text{:Action}} \quad \dfrac{q\text{:Question}}{\textbf{raise}(q)\text{:Action}} \quad \dfrac{p\text{:Prop}}{\textbf{instruct}(p)\text{:Action}}$

That is, dialogue participants may either raise questions (put them on QUD), respond to questions (which are maximal in QUD) or give an instruction to the other dialogue participant. We are trying here the experiment of doing as much as possible in terms of raising or responding to questions.

Transitions between information states which are occasioned by a dialogue contribution are defined in terms of a restricted set of operations. Again, this is probably more restricted than is ultimately needed, but we want to start small and then see what motivation there is for making additions. The operations we have used in this coding are given in (8).

(8) Stack: pushRec, popRec (push and pop stack in record)
Set: addRec (add to set in record)

The following example shows how updates change the information state during a dialogue. (9) shows the information state before utterance **U4** in the Autoroute dialogue presented in full in Appendix A. (10) shows the utterance itself and the accompanying updates. (11) shows the information state after the updates.

(9)
$$\begin{bmatrix} \text{A} = \begin{bmatrix} \text{PRIVATE} = \begin{bmatrix} \text{AGENDA} = \left\langle \begin{array}{l} \textbf{raise}(\textit{'Where does B want to start?'}) \\ \textbf{raise}(\textit{'Where does B want to go?'}) \\ \textbf{raise}(\textit{'What time does B want to go?'}) \\ \textbf{raise}(\textit{'Does B want the quickest or} \\ \textit{shortest route?'}) \end{array} \right\rangle \end{bmatrix} \\ \text{SHARED} = \begin{bmatrix} \text{BEL} = \left\{ \begin{array}{l} \textit{'B wants a route from A'} \\ \textit{'B has A's attention'} \\ \textit{'A has B's attention'} \end{array} \right\} \end{bmatrix} \end{bmatrix} \\ \text{B} = \begin{bmatrix} \text{PRIVATE} = \begin{bmatrix} \text{BEL} = \left\{ \textit{'B wants assistance'} \right\} \end{bmatrix} \\ \text{SHARED} = \begin{bmatrix} \text{BEL} = \left\{ \begin{array}{l} \textit{'B wants a route from A'} \\ \textit{'B has A's attention'} \\ \textit{'A has B's attention'} \end{array} \right\} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

(10) **U4 [A]: Where would you like to start your journey.**

popRec(A.PRIVATE.AGENDA)
pushRec(A.SHARED.QUD, '*Where does B want to start?*')
pushRec(B.SHARED.QUD, '*Where does B want to start?*')
pushRec(B.PRIVATE.AGENDA, **respond**('*Where does B want to start?*'))

$$
(11)\quad \begin{bmatrix}
\text{A} = \begin{bmatrix}
\text{PRIVATE} = \begin{bmatrix} \text{AGENDA} = \left\langle \begin{array}{l} \textbf{raise}(\text{'}\textit{Where does B want to go?}\text{'}) \\ \textbf{raise}(\text{'}\textit{What time does B want to go?}\text{'}) \\ \textbf{raise}(\text{'}\textit{Does B want the quickest or} \\ \textit{shortest route?}\text{'}) \end{array} \right\rangle \end{bmatrix} \\
\text{SHARED} = \begin{bmatrix} \text{BEL} = \left\{ \begin{array}{l} \text{'}\textit{B wants a route from A}\text{'} \\ \text{'}\textit{B has A's attention}\text{'} \\ \text{'}\textit{A has B's attention}\text{'} \end{array} \right\} \\ \text{QUD} = \left\langle \text{'}\textit{Where does B want to start?}\text{'} \right\rangle \end{bmatrix}
\end{bmatrix} \\
\text{B} = \begin{bmatrix}
\text{PRIVATE} = \begin{bmatrix} \text{BEL} = \left\{ \text{'}\textit{B wants assistance}\text{'} \right\} \\ \text{AGENDA} = \left\langle \textbf{respond}(\text{'}\textit{Where does B want to start?}\text{'}) \right\rangle \end{bmatrix} \\
\text{SHARED} = \begin{bmatrix} \text{BEL} = \left\{ \begin{array}{l} \text{'}\textit{B wants a route from A}\text{'} \\ \text{'}\textit{B has A's attention}\text{'} \\ \text{'}\textit{A has B's attention}\text{'} \end{array} \right\} \\ \text{QUD} = \left\langle \text{'}\textit{Where does B want to start?}\text{'} \right\rangle \end{bmatrix}
\end{bmatrix}
\end{bmatrix}
$$

Note that the SHARED fields are not the same for the two dialogue participants. They may have different views about what has been established in the dialogue and what is currently under discussion. Such differences may arise because of genuine misunderstanding. But they may also arise because of the general dialogue strategy pursued by the participants which lead to mismatches which would not be intuitively construed as misunderstandings.

For a more elaborate discussion of the Cooper-Larsson information state and its relation to dialogue moves and an "optimistic" grounding strategy, see [7]. The theory has also been extended and implemented in a dialogue system, described in [3].

3.2 Scheme 2: The Poesio-Traum model of Information States

The second model of information states is based on the dialogue model of Poesio and Traum [17, 18]. One of the central concerns of this work, which builds upon previous work by Traum [20], is the GROUNDING process, by which common ground is established [4, 21]. Poesio and Traum view the public information state as including both the material that has already been grounded, indicated as G here, and of the material that hasn't yet been grounded; the ungrounded part consists of a specification of the current 'contributions,' or DISCOURSE UNITS (DUs), as they are called in [21].

As in the case of the notion of information state developed by Cooper and Larsson, the information state of each agent is explicitly represented in the feature-based representation. A difference, though, is the representation of individual DUs representing information introduced into the dialogue but not yet considered shared. G and each DU will be represented as a separate record within each participant's record. Also, the private information about an agent's mental state is not given a separate record, like `private` scheme 1, but represented as individual fields in the record for the participant.[3] In terms of private information, we generally represent two types. First, a list of ungrounded DUs (UDUs) which represents which of the DUs are on the way to being grounded. Secondly, the participants intentions to act related to the dialogue. This is currently represented as an ordered list of prioritised actions, as in (12)

(12) A: < Ask for start place (GET-SP),
Ask for destination (GET-DEST),
Ask for start time (GET-ST),
Ask if quickest or shortest route desired (GET-ROUTE-TYPE)>

The record for each participant is thus of the type shown in (13).

$$(13)\quad \begin{bmatrix} \text{G} & : & \textit{PT-record} \\ \text{DU}_1 & : & \textit{PT-record} \\ \ldots & & \\ \text{DU}_n & : & \textit{PT-record} \\ \text{UDUs} & : & \textit{List} \\ \text{INT} & : & \textit{List} \end{bmatrix}$$

A second difference between the Poesio-Traum information states and that of Cooper-Larsson, described in the previous section, is the information types within the modalities. In the Poesio-Traum model there are several kinds of information kept in the shared (G) and semi-public (DUs) part of a participants information state. First, an explicit history of the dialogue acts[4] that have been performed. For simplicity, we represent that here as a list, abbreviated DH. Next we represent the social commitments, or obligations of the agents. These kinds of commitments come in two forms, depending on whether the agent is committed to a fact being the case, or to act in a particular way. We term the former SOCIAL COMMITMENTS TO A PROPOSITION, abreviated as SCP in the information state. The latter we call "Obligations", abbreviated as OBL. Also, we have a set of OPTIONS, abbreviated as OPT, representing actions which no agents have been obliged to perform, but which have been explicitly discussed as possibilities. Thus, each DU, as well as G will be a record of the type shown in (14) (abbreviated PT-record in (13)).

$$(14)\quad \begin{bmatrix} \text{DH} & : & \textit{List} \\ \text{OBL} & : & \textit{List} \\ \text{SCP} & : & \textit{List} \\ \text{OPT} & : & \textit{List} \end{bmatrix}$$

The obligations that are part of OBL are generally to perform a particular type of dialogue action, (e.g., 'address' or 'answer') with pointers to the relevant moves in the DH. An example is given in (15), which indicates that participant A has an obligation to answer Move 2, while participant B has obligations to answer Move 3 and to address Move 1. Obligations and commitments can also be conditional on particular actions being performed in the future.

(15) < A ANSWER 2, B ANSWER 3, B ADDRESS 1 >

To summarize, each information state will be of the type in (16).

(16)

$$\left[\begin{array}{ll} \text{A:} & \left[\begin{array}{ll} \text{G:} & \left[\begin{array}{l} \text{OBL: } < \ldots > \\ \text{DH: } < \ldots > \\ \text{SCP: } < \ldots > \\ \text{OPT: } < \ldots > \end{array}\right] \\ \text{INT: } < \ldots > & \\ \text{DUi: } \ldots & \\ \text{UDUS: } < \text{DUi}, \ldots > & \end{array}\right] \\ \text{B:} & \left[\begin{array}{ll} \text{G:} & \left[\begin{array}{l} \text{OBL: } < > \\ \text{DH: } < > \\ \text{SCP: } < \ldots > \\ \text{OPT: } < \ldots > \end{array}\right] \\ \text{INT: } < \ldots > & \\ \text{DUi: } \ldots & \\ \text{UDUS: } < \text{DUi}, \ldots > & \end{array}\right] \end{array}\right]$$

To see how this notion of information state applies to representing the effects of utterances, consider the same utterance used to exemplify the Cooper and Larsson approach. (17) shows the information state after utterance U4. The effect of a new utterance is to create a new DU (DU4), which becomes part of the information state of both agents. The main difference between the information states of the agents in this case is that B has the intention to get a route from Malvern to Edwinstowe, whereas A has the intentions to get the information he needs to address that request.

(17) **U4 [A]: Where would you like to start your journey.**

$$\left[\begin{array}{ll} \text{A:} & \left[\begin{array}{ll} \text{G:} & \left[\begin{array}{l} \text{OBL: } < \text{B UNDERSTANDING-ACT 4B, A ADDRESS 3C} > \\ \text{SCP: } < \text{B WANTS A ROUTE} > \end{array}\right] \\ \text{INT:} & < \begin{array}{l} \text{GET(SP), GET(DEST), GET(ST), GET(ROUTE-TYPE),} \\ \text{GIVE B ROUTE(SP,DEST,ST,ROUTE-TYPE)} \end{array} > \\ \text{DU4A:} & \left[\begin{array}{l} \text{OBL: } < \text{B ANSWER 4} > \\ \text{DH: } < \text{4: INFO-REQUEST} > \end{array}\right] \end{array}\right] \\ \text{B:} & \left[\begin{array}{ll} \text{G:} & \left[\begin{array}{l} \text{OBL: } < \text{B UNDERSTANDING-ACT 4B, A ADDRESS 3C} > \\ \text{SCP: } < \text{B WANTS A ROUTE} > \end{array}\right] \\ \text{INT:} & < \text{GET A ROUTE FROM MALVERN TO EDWINSTOWE} > \\ \text{DU4B:} & \left[\begin{array}{l} \text{OBL: } < \text{B ANSWER 4} > \\ \text{DH: } < \text{4: INFO-REQUEST} > \end{array}\right] \end{array}\right] \end{array}\right]$$

(18) shows the information state resulting from B's answer in U5. This results in DU4 being grounded, i.e., added to G. B's obligation to answer 4 is moved to G and stays there until his action is grounded. B commits himself to the belief that the starting point is Malvern. (We only show A's info state for brevity.)

(18) **U5 [B]: Malvern.**

$$\left[\text{A:}\left[\begin{array}{ll} \text{G:} & \left[\begin{array}{l}\text{OBL: < A \textsc{understanding-act} DU5B, B \textsc{answer} 4, A \textsc{address} 3C>}\\ \text{SCP: <B \textsc{wants a route} >}\end{array}\right]\\ \text{INT:} & <\begin{array}{l}\text{\textsc{get}(SP), \textsc{get}(DEST), \textsc{get}(ST), \textsc{get}(ROUTE-TYPE),}\\ \text{\textsc{give} B \textsc{route}(SP,DEST,ST,ROUTE-TYPE)}\end{array}> \\ \text{DU5B:} & \left[\begin{array}{l}\text{OBL: < A \textsc{address} 5B >}\\ \text{SCP: <B \textsc{believes} SP = \textsc{Malvern} >}\\ \text{DH: < 5A: \textsc{answer}, 5B: \textsc{assert} >}\end{array}\right]\end{array}\right]\right]$$

As well as the updates on the individual aspects of the information state, we will also want a more complex *merger* of DU modalities, to represent the grounding process. The basic idea is that when a DU is acknowledged, all of the information from that DU is merged into G. The immediate effect will be to merge the various fields in the updated record. Other effects will involve removing some items from fields, e.g., when noticing that an obligation has been fulfilled. We will represent this merging of G with information from another DU as in (19).

(19) G+= DU

4 Annotation

Annotating for information states is not a particularly quick business, even given the simple information states we have in our current examples (and it requires LATEXnical stamina). We have investigated two different ways of coping with this problem, one using an annotation tool called *TranScript* and the other using the *Thistle Diagram Editor.*

4.1 Annotation as scripting: TranScript

TranScript is a coding tool we have developed intended for the kind of relatively complex annotation necessary when annotating transcriptions (usually of spoken dialogue) with information state updates. In this kind of annotation, simple tags as those used in e.g. part-of-speech tagging (noun, verb etc.) are not sufficient. An information state update annotation (henceforth "update annotation") may have several arguments, i.e. the participant affected, what part of the information state is being updated (private beliefs, agenda etc.), type of update (add, push etc.), and additional arguments such as propositional content and action type. A parser and and a LaTeX generator for TranScript annotation has been implemented in SICSTUS Prolog.

The basic idea behind TranScript is that the annotation can be seen as a kind of script, which is a variation on the idea of tagging with logic programs as in TagLog [15]. The major difference is that the ordering of the annotation clauses are important. In TagLog, each clause contains a reference to a stretch of transcribed text in the transcription file. For example, in the clause `part_of_speech(34-35,noun).`, the range 34-35 refers to the word between positions 34 and 35 in the transcription. In TranScript, this reference is indicated by the ordering of clauses. Each update implicitly refers to the latest range of transcription indicated above it. For example, in the following example the updates refer to the range 157-209.

(20)
```
# range(157-209).

label(q8, "Where does B want to start?").

# update([ popRec(a*private*agenda),
           pushRec(a*shared*qud, $q8),
           pushRec(b*shared*qud, $q8),
           pushRec(b*private*agenda, respond($q8))
        ]).

# print_state.
```

TranScript contains elements of logic programming. The $ sign indicates a label, and labels are declared with the `label` predicate as in the example above. The use of labels provides a simple way to refer to propositional contents, actions etc. in annotation clauses. A typical use of TranScript annotation files is to parse them and translate them into a sequence of information states, which then can be used to give a LaTeXversion of the transcription with information states and updates indicated as in (9)-(11).

4.1.1 TranScript Commands

There are two kinds of commands in TranScript: order-independent (purely declarative) commands and order-dependent commands, or *script commands*. The declarative commands include `initial` and *label*, and the script commands are any defined operations or moves, `update`, `range`, `print_state` and `comment`. Script commands must always be preceded by the # symbol.

Update operations are used to actually update the previous information state to produce a new one. The `update` command takes as argument a list of updates, which consist of an update operator (or a move) and its arguments. The type of update operators available depend on the notion of information state; if the information state is a set of propositions, a typical update would be `add($p12)`, where `add` is an update operator and `$p12` is a label for a proposition (whatever that might be). The order of the updates list is important, since the information state will be updated with each operation one at a time in the order they are given. For example, if two things are to be pushed onto a stack, the order of the operations will determine the resulting stack.

(21)

Datatype	**Operators**
Set	add, del(ete)
Stack	push, pop
Record	addField, get_valueRec, set_valueRec, pushRec, popRec, addRec, delRec, peRec
DRS	get_valueDRS, set_valueDRS, mergeDRS

4.1.2 Parsing TranScript files

The TranScript implementation consists of two main modules: a parser and an output generator. These modules communicate via a set of instructions which can be regarded as an "inflated" version of the TranScript instructions, where information states have been filled in and labels have been replaced with their corresponding values. The parser reads the TranScript instructions and updates and keeps track of the current information state, successively updating it. The "translations" done by the default TranScript parser can be summed up in the following table:

(22)

TranScript command	Inflated TranScript command(s)
any operator or move U	`print_update(`U_i`)` where U_i is the inflated version of U
`update(`Us`)`	`print_updates(`Us_i`); print_state(`IS`)` where Us_i is the inflated version of Us and IS is the current infostate
`print_state`	`print_state(`IS`)` where IS is the current infostate
`range(`R`)`	`print(`S`)` where S us the string in range R of the transcription file
`print(`S`)`	`print(`S`)`
`comment(`C`)`	`print_comment(`C`)`

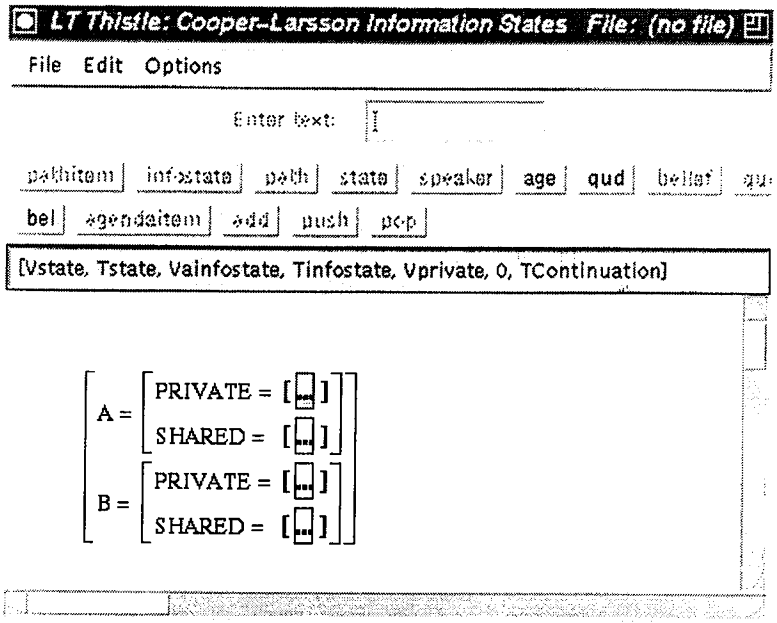
4.1.3 Generating output

The LaTeXgenerator takes an inflated TranScript file and produces a LaTeXfile, following the specified list of instructions. Various conventions are used to increase readability; for example, sentences are printed in *italics*, record labels are printed in SMALL CAPS, and actions are printed in **bold** style. It is also possible to implement output generators for other kinds of output (ASCII text, HTML etc), as long as they understand the inflated TranScript instruction set.

4.2 Annotation Using the Thistle Diagram Editor

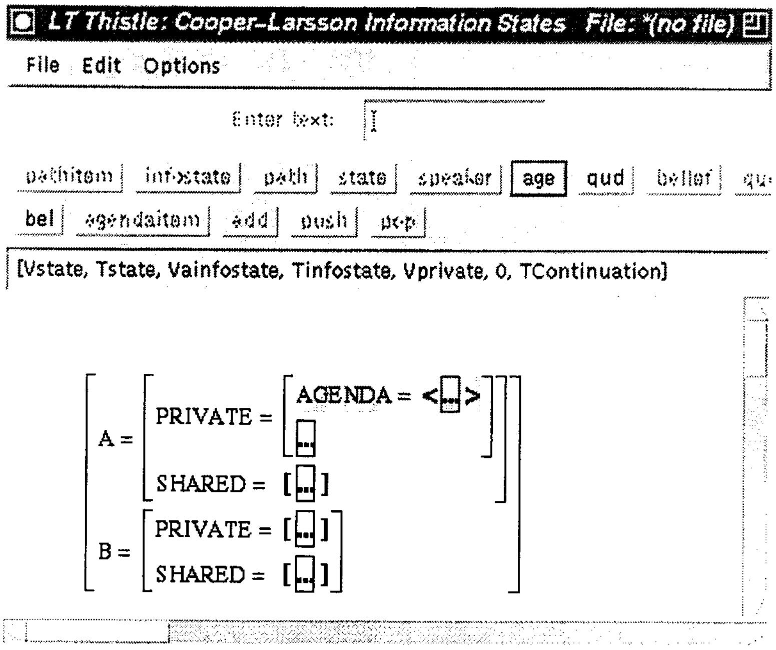
Thistle is a parameterizable display engine and editor for diagrams which allows the inclusion of interactive diagrams within Web pages. See `http://www.ltg.ed.ac.uk/software/thistle` for full information on the program and for a number of demos. We have adapted Thistle to the task of annotating ISs in dialogue.

Diagram specifications for ISs in both the Cooper-Larsson and Poesio-Traum models have been written, allowing the structures to be displayed and edited. In order to illustrate this, the diagram below shows an 'empty' Cooper-Larsson IS in Thistle:



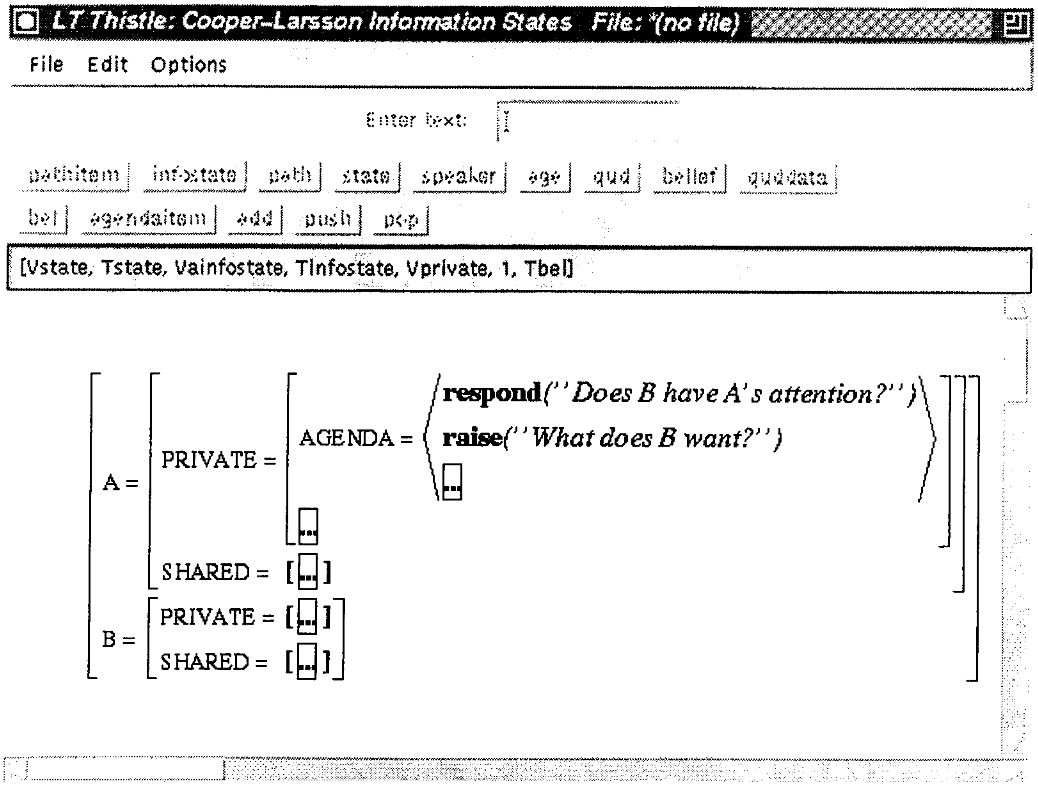
(23)

This diagram class is designed as an Attribute-Value Matrix, a standard way of representing linguistic (and other) information. Here three dots inside a box indicate areas where a diagram can be expanded, and in this case the user has clicked on the value of the path A=PRIVATE, with the result that the box is shaded and the possible types which can expand the diagram are enabled in the menu bar at the top half of the window. The options (defined by the theory, of course), are *age* (to expand the AGENDA field), *qud* (expand QUESTION UNDER DISCUSSION), and *bel* (expand BELIEFS). If the user selects *age*, the result is:



(24)

The diagram is expanded appropriately, and if the user now clicks in the expansion box in the value of AGENDA, selects *agendaitem* (the only choice) and adds the appropriate text, the result is:



(25)

The appearance of the text (the bold and italic fonts and the small capitals for the attribute names) are all handled automatically by the diagram class specification. Similarly, the precise layout of the AVM is done by the program – all the user has to do is supply the appropriate paths and values.

It is thus possible to use Thistle to produce annotations in the form of ISs, and including a representation of the updates in the diagram class is a simple matter. However, doing large-scale annotation in this way is clearly time-consuming, and it is likely that this method would only be useful to produce small illustrations.

5 Discussion

Our pilot work (reported in [8]) involved using the annotation schemes discussed above to study the dialogue in Appendix A; this involved several annotations and subsequent revisions. In this section we are going to discuss our preliminary conclusions about the methodology and some empirical issues raised by this work.

We do feel that the methodology we are developing could be useful both (i) for people who are interested in studying dialogue acts either from an empirical perspective or by looking in more detail at the formal differences between systems; and (ii) for people who are building systems, who could just come up with a characterization of their information states without worrying about formal details, a characterization of the updates each dialogue act performs, and then use the tool to check that their definitions of dialogue acts behave as intended.

There are however some potential problems to be considered. First of all, since the notation does not wear its semantics on its sleeves, more detailed comparisons between theories will involve either more detailed annotations, or spelling out the interpretation of primitives such as intentions and obligations, or both. For example, we have been investigating the differences between a model based on obligations and a model based on questions under discussions; but such differences cannot be revealed as long as the only constraint we impose on the fields is that their values are stacks.

Secondly, it has become even clearer to us that annotating for information states is not suitable for large-scale annotation: both because it is time consuming, and because it is even more difficult to agree on the composition of the information state of an agent than it is to agree on which dialogue act has been performed. It definitely seems to be the case that this type of annotation should be used in the preliminary phases of an annotation work, to come up with a taxonomy of dialogue acts that appears to have adequate coverage and matches the operations that the system has to perform; subsequent, large scale annotation can then be done in terms of atomic labels.

For future work, we plan to integrate the annotation tools into the TRINDI Dialogue Move Engine toolkit (see [19], [16]). This will make it possible to define dialogue moves and information state update rules in terms of update operations, and to annotate using these moves and rules. The toolkit will also enable visualization (and possibly editing) of information states using Thistle.

Acknowledgments

This work was supported by the TRINDI (Task Oriented Instructional Dialogue) project, EU TELEMATICS APPLICATIONS Programme, Language Engineering Project LE4-8314. We are grateful to the Speech Research Unit of the Defence Evaluation and Research Agency, Malvern, UK, for making the Autoroute dialogues available to the Trindi project.

MASSIMO POESIO, COLIN MATHESON
Human Communication Research Centre
University of Edinburgh
2 Buccleuch Place Edinburgh EH8 9LW, Scotland, UK

ROBIN COOPER, STAFFAN LARSSON
Department of Linguistics
Göteborg University
Box 200, SE-405 30 Göteborg, Sweden

DAVID R TRAUM
UMIACS
A. V. Williams Building
University of Maryland
College Park, MD 20742 USA

Notes

[1]Additional constraints can also be imposed by the user - e.g., minimizing time, or toll cost, etc. We will ignore these constraints here.

[2]This is most likely an oversimplification; it will probably be necessary to have a separate field for goals.

[3]We do this for two reasons. First, just to avoid the need for an extra record indirection when coding, and secondly, to be closer to the DRT-based theory in [17, 18], which relied on DRT accessibility relations. For the purposes of this record-based model of information state, there is nothing wrong with viewing these other aspects of the mental state as belonging to a subrecord for the modality `private`, so as to conform to the specification in (4).

[4]In the Poesio-Traum model, the DRI dialogue acts ([2], [9]) are used.

References

[1] D. Albesano, P. Baggia, M. Danieli, R. Gemello, E. Gerbino, and C. Rullent. A robust system for human-machine dialogue in a telephony-based application. *Journal of Speech Technology*, 2(2):99–110, 1997.

[2] J. Allen and M. Core. DAMSL: Dialogue act markup in several layers. Draft contribution for the Discourse Resource Initiative, October 1997.

[3] P. Bohlin, R. Cooper, E. Engdahl, and S. Larsson. Information states and dialogue move engines. In J. Alexandersson, editor, *IJCAI-99 Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, 1999.

[4] H. H. Clark and E. F. Schaefer. Contributing to discourse. *Cognitive Science*, 13:259 – 94, 1989.

[5] R. Cooper. Information states, attitudes and dependent record types. In *Proceedings of ITALLC-98*, 1998.

[6] R. Cooper. Mixing situation theory and type theory to formalize information states in dialogue exchanges. In *Proceedings of TWLT13/Twendial '98: Formal Semantics and Pragmatics of Dialogue*, 1998.

[7] R. Cooper and S. Larsson. Dialogue moves and information states. In *Proc. of the Third IWCS*, Tilburg, 1999.

[8] R. Cooper, S. Larsson, C. Matheson, M. Poesio, and D. Traum. Coding instructional dialogue for information states. deliverable D1.1, TRINDI, 1999.

[9] Discourse Resource Initiative. Standards for dialogue coding in natural language processing. Report no. 167, Dagstuhl-Seminar, 1997.

[10] J. Ginzburg. Resolving questions, i. *Linguistics and Philosophy*, 18(5):567–609, 1995.

[11] J. Ginzburg. Resolving questions, ii. *Linguistics and Philosophy*, 18(6):567–609, 1995.

[12] J. Ginzburg. Dynamics and the semantics of dialogue. In Jerry Seligman and Dag Westerståhl, editors, *Logic, Language and Computation, Vol. 1*, volume 1. CSLI Publications, 1996.

[13] J. Ginzburg. Clarifying utterances. In J. Hulstijn and A. Niholt, editors, *Proc. of the Twente Workshop on the Formal Semantics and Pragmatics of Dialogues*, pages 11–30, Enschede, 1998. Universiteit Twente, Faculteit Informatica.

[14] B. J. Grosz and C. L. Sidner. Attention, intention, and the structure of discourse. *Computational Linguistics*, 12(3):175–204, 1986.

[15] T. Lager. *A Logical Approach to Computational Corpus Linguistics.* PhD thesis, Detp. of Linguistics, Göteborg Universtity, 1995.

[16] S. Larsson, P. Bohlin, J. Bos, and D. Traum. Coding instructional dialogue for information states. deliverable D2.2, TRINDI, forthcoming.

[17] M. Poesio and D. Traum. Conversational actions and discourse situations. *Computational Intelligence*, 13(3):309–347, 1997.

[18] M. Poesio and D. Traum. Towards an axiomatisation of dialogue acts. In J. Hulstijn and A. Nijholt, editors, *Proc. of the Twente Workshop on the Formal Semantics and Pragmatics of Dialogues*, pages 207–222, Enschede, 1998. Universiteit Twente, Faculteit Informatica.

[19] D. Traum, J. Bos, R. Cooper, S. Larsson, I. Lewin, C. Matheson, and M. Poesio. Coding instructional dialogue for information states. deliverable D2.1, TRINDI, forthcoming.

[20] D. R. Traum. *A Computational Theory of Grounding in Natural Language Conversation.* PhD thesis, University of Rochester, Department of Computer Science, Rochester, NY, July 1994.

[21] D. R. Traum and E. A. Hinkelman. Conversation acts in task-oriented spoken dialogue. *Computational Intelligence*, 8(3), 1992. Special Issue on Non-literal Language.

A Autoroute dialogue 127

U1 [A]: Welcome to the Route Planning Service.
U2 [A]: How can I help you.
U3 [B]: A route please.
U4 [A]: Where would you like to start your journey.
U5 [B]: Malvern.
U6 [A]: Starting in Great Malvern.
U7 [B]: Yes.
U8 [A]: Where would you like to go.
U9 [B]: Edwinstowe.
U10 [A]: Edwinstowe.
U11 [B]: Yes.
U12 [A]: Please wait.
U13 [A]: Is that Edwinstowe in Nottingham.
U14 [B]: Yes.
U15 [A]: What time would you like to make your journey.
U16 [B]: Six p.m.
U17 [A]: Leaving at six p.m.
U18 [B]: Yes.
U19 [A]: Would you like the quickest or the shortest route.
U20 [B]: Quickest.
U21 [A]: Please wait while your route from Malvern to Edwinstowe is calculated.
U22 [A]: The quickest route is one hundred and thirteen miles and will take two hours eight minutes.
U23 [A]: Would you like me to send the instructions to you.
U24 [B]: No.

U25 [A]: Do you require any further information now.
U26 [B]: No.
U27 [A]: Can I have your name please.
U28 [B]: Mr Smith
U29 [A]: Mr Smith
U30 [B]: Yes
U31 [A]: And your location please.
U32 [B]: T 43
U33 [A]: T 43
U34 [B]: Yes.
U35 [A]: OK.
U36 [A]: Thank you for calling.
U37 [A]: Goodbye.

The Structure of Task-oriented Dialogue and the Introduction of New Objects*

Hannes Rieser

Bielefeld University
Faculty of Linguistics and Literary Studies and
Cognitive Science Research Unit SFB 360 "Situated Artificial Communicators"
Projects B3 and C4

Contents

*I felt very honoured to give the talk on which this paper is based as an invited speaker in the section *Dialogue Analysis* at the renowned *Amstelog99*. I want to thank the organizers Jan van Kuppevelt, Noor van Leusen, Robert van Rooy, and Henk Zeevat for their invitation and the *Amstelog99* participants for their comments on my talk. I am especially grateful to Hans Kamp for commenting upon my epistemically based perspective on anaphora and to Jan van Kuppevelt and Joris Hulstijn for their comments on dialogue structure and the "track" idea. This paper could not have been produced without the help of various good-willed people: Peter Kühnlein did the LaTeXing. Uwe Weigang, Guido Heumer, and Josef Meyer-Fujara contributed various illustrations. Thanks to all of them.

1 Introduction

The following paper deals with two topics, the structure of task-oriented dialogue and the introduction of new objects into dialogue by one of the agents. The discussion of the structure of task-oriented dialogue is based on a corpus of dialogues (transcripts, speech, video-films of agents' actions, eye-movement data, see *Corpus*; Clermont et al. (1995); Pomplun et al. (1998)) based on the construction of a little toy-airplane (see Fig. 1) and tries to take account of what the data show, irrespective of idealisations which might them make fit smugly into one of the currently favoured approaches to dialogue. This opens up a new perspective on the ways agents use to organise the information flow. Above all, the role of coordination and the evolution of common ground for the step-by-step progress of dialogues is worked out. Common ground is anchored in the dialogue situation. Different settings for task-oriented dialogues, especially those without visual contact between the agents show that common ground is something actively worked at. Above all it is based on mutual beliefs having high plausibility for the agents. Looked at from the empirical side, common ground has to keep track of expressions, matters of reference and interpretation, properties of situations, actions done, and the history of the verbal exchanges carried out. Ideally, all these aspects should go into the set up of dialogue models.

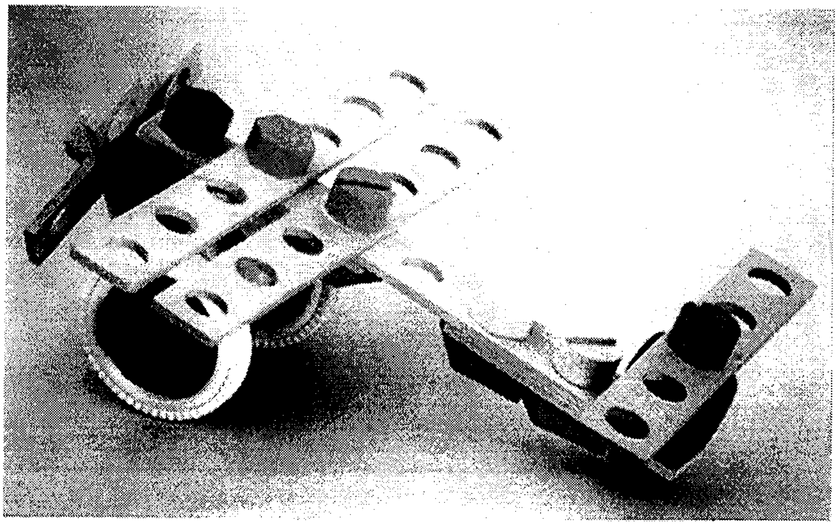


Figure 1: Toy-airplane

Empirical data clearly show that the rise of beliefs and the agents' temporarily maintaining them is a central mechanism for the progress of a dialogue, determining e.g. next contributions. This holds good for the agents' introduction of new objects into dialogue as well. Agents have to be sure that objects have been introduced, especially in non-face-to-face conditions, before they can anaphorically refer to them. Therefore they spend a lot of energy on reliable object introduction and use elaborate techniques to heighten chances of success. This indicates that theories of anaphora in dialogue have to be sensitive to belief formation and common ground evolution.

2 From Non-Dialogue to Dialogue

Non-logically based research on discourse has concentrated on various themes such as the introduction of discourse topics, the structure of paragraphs, the incremental generation of sequences of sentences, the appropriate specification of temporal, causal or other relations among the events depicted. All of these involve in one way or other the resolution of anaphora. Seen from the perspective of descriptive linguistics, the resolution of anaphora has been one of the main contributions of Dynamic Semantics to the description of non-dialogical discourse. Thus Discourse Representation Theory (DRT, Kamp and Reyle (1993)) or Dynamic Predicate Logic (DPL, Groenendijk and Stokhof (1991)) investigated the familiar pattern of indefinite NPs resumed anaphorically as in (1)

(1) *A man walks in the park. He whistles.*

DRT provides a DRS for the first sentence via a construction algorithm and then adds the information of the second sentence to the first. The discourse referent (DR) introduced for the indefinite NP *a man* is used as a hook for the DR standing proxy for the pronoun *he*. The linking of both DRs is achieved by an accessibility relation. Intuitively, the DR for *a man* binds the DR for *he*. In DPL and in paradigms based on DPL like Chierchia's theory of Dynamic Meaning (Chierchia, 1995), a similar effect is achieved: The binding potential of the existential quantifier standing for the indefinite NP is "stretched" and the free variable standing for *he* is bound using something like axiom (2):

(2) $\exists x\phi \wedge \psi \simeq \exists x[\phi \wedge \psi]$

. Our initial description of the aims of non-logically based research in discourse structure already showed that there is more to discourse than anaphorical links. Hence, it does not come as a surprise that proposals extending the original set up of Dynamic Semantics soon were made. They first concerned non-dialogues and had different aims in focus. E. g. Asher and cooperators worked on the development of SDRT and dealt with structural relations in discourse like narration, explanation and elaboration (Asher, 1993; Lascarides and Asher, 1993). These were tied up with so-called rhetorical functions. Pursuing a somewhat different line, Kamp (1990) tried to provide an explanation for the use of antecedentless deictic NPs as in

(3) *That man is a cocaine dealer.*

Kamp's 1990 paper considers speakers' and hearers' beliefs and is thus already on the verge of dialogue description. From the historical point of view, DRT's step towards intention and belief-based dialogue models seemed to be small then, but it was not made, at least not officially. The main thrust of Dynamic Semantics for dialogue started from the principal technique established with respect to non-dialogue texts: resolution of anaphora. However, several papers and talks presented during the last three years or so (see Dekker (1997); Dekker and van Rooy (1998); Poesio (1998)) concentrated on the topic of so-called "cross-speaker anaphora". Similarly, Asher and cooperators generalized SDRT's rhetorical functions to dialogue and also started to investigate divergent discourse relations like corrections, counterevidence, and contradiction (Asher, 1998) as in:

(4) a. A: *They gave Peter the new computer.*

b. B: *No, they gave [John]$_F$ the new computer.*

Example (4) shows that work in Dynamic Semantics gradually shifted from speaker-free discourse to dialogue. Passing from (1) to (5), the dialogue version of (1), seems to be simple. All you seem to have to do is to fit in speakers A and B and divide up the discourse portions among them, a man a contribution each:

(5) a. A: *A man walks in the park.*

b. B: *He whistles.*

At first sight, anaphora resolution in (5) seems to work as in (1). What I want to show in discussing empirical material is that there is a giant step from (1) to (5). As a consequence, it seems to me that if theories were developed capturing the use of anaphora in (5) they could well shed light on data as in (1) rather than vice versa. This is quite opposite to the research strategy from speaker-free discourse to dialogue as it has been pursued in much current work.

3 How Do the Pieces of a Dialogue Fit Together?

In order to see the difference between (1) and (5) we can start[1] with the question "How are the contributions of A and B in (5) bound together to make up a whole in a fairly natural situation in which A's and B's contributions are anchored?". A somewhat opposite scenario to the preferred natural one would look as follows: We have two tokens uttered in a distance, perhaps at different times, recorded by chance with the same device, perhaps some heavenly microphone. We would not consider that a dialogue, would we?

There is of course a minor handicap to our plan to embed (5) in a natural situation. (5) is not a full dialogue, the prototypical entries into and exits from it are missing. This is an important aspect, which cannot be dealt with here, however. Nevertheless, we can ask leading questions with respect to (5) which hopefully further our hermeneutic enterprise:

(a) How are the consecutive utterances/sentences related to each other?

[1]For similar observations with respect to Japanese cf. Kawamori and Shimazu (1999), this volume.

(b) How are the agents' consecutive acts related to each other?

(c) What are the principles on which the alternation of agents' acts is based?

We will now discuss (a), (b), (c) in turn.

3.1 Discussion of question (a) "How are the consecutive utterances/sentences related to each other?"

Question (a) can be dubbed "structure-oriented". It has been investigated wrt sentences in the dynamic paradigms DRT, SDRT and DPL. (b) and (c) imply that we move from considering abstract textual structure to discussing agents' activities. Investigations concerning question (a) have a longer tradition in linguistic scholarship, also, (a) seems to be easier to answer than the other two.

The folklore tradition we can draw on is that consecutive utterances of different agents can be bound together by specific patterns sometimes called "adjacency pairs". More prototypical exemplars of such pairs are for example question–answer, directive–reply or summons–response. Adjacency pairs can be chained together or embedded into each other. They can hence be used as a device to build up larger dialogue structures. Observe that (5) is *prima facie* not a good example of an adjacency pair. Contrary to what happens in question–answer pairs, B's contribution to (5) seems in no way necessitated by A's utterance. At first glance, it looks like a mere addition. Indeed it is hard to explain, just why B started to talk at all. We will return to this topic in greater detail in our section 6, "Constructive Deconstruction: Re-situating the Walker-in-the-Park", pointers to it will, however, be set here and there throughout this paper.

3.2 Discussion of question (b) "How are the agents consecutive acts related to each other?"

Question (b) is clearly related to agents' acts. I do not know of many elaborated attempts to find an answer to it. An exception is H. H. Clark's work, especially his 1996 book. It is based on research in different fields, above all, in cognitive psychology, discourse analysis and analytical philosophy of language. Taking the action perspective seriously, Clark argues that agents establish "joint projects" in order to solve problems such as the exchange of goods or the transfer of information. These joint projects are based on talk exchange. In pursuing their business, agents coordinate and negotiate. Jointness is the all-important feature in many of our mundane transactions. The joint-project idea generalizes fairly easily to contexts where straightforward adjacency pairs like question–answer or directive–reply are used. Our dialogue version of (1), (5), cannot, however, be subsumed under the notion of joint project without considering special types of situations.

Coordination in Clark's sense is first of all connected with the production and the understanding of single utterances. Utterances are conceived of as a series of joint projects from phonetic discrimination to up-take of the speaker's proposal. This series is called "action ladder" for obvious reasons. Agents' negotiations ranging from phonetic discrimination to uptake of proposal found in our empirical data yield ample support for the idea of action ladders. In Fig. 2 the parallel activities of speaker and addressee are shown. It presents a illustration of the main ideas involved.

Speaker	**Addressee**
Execution of behaviour	**Attending to behaviour**
Types of actions	
summons	answer
gazing	acknowledging by gaze
restart of turn	attention to signal
Presentation of signal	**Identification of signal**
Types of actions	
production of turn	question
(6) (Clark, 1996, p. 221)	Roger: *now, - um do you and your husband* ***have a j- car.*** Nina: ***have a car?*** Roger: *yeah.* Nina: *no.*
(7) (Clark, 1996, p. 240)	Morris: *so I wrote off to. Bill, . uh who ((had)) presumably disappeared by this time, certainly, a man called* ***Annegra?*** *-* June: *yeah,* ***Allegra*** Morris: ***Allegra****, uh replied,. uh and I. put. two other people, who'd been in for. the BBST job. with me* [continues]
Signalling that p **Meaning that p**	**Recognition that p** **Understanding that p**
Types of actions	
production of expression	assertion of understanding: *I see, yes, mhm, yeah, right* paraphrase, verbatim repetition, completion of speaker's utterance, initiation of next turn
Proposal of joint project	**Consideration of joint project**
Types of actions	
(8) (Clark, 1996, p. 318)	Jane: *hello, is Miss Pink in.* Kate: *well, she's in, but she's engaged at the moment, who is it?*

Figure 2: Clark's "action ladder": Speaker's and hearer's parallel activities

The speaker executes behaviour to which the addressee attends. Execution of behaviour can *inter alia* (see examples in Fig. 2) be realized by gaze.

The addressee will attend to the behaviour and can—in our example—also acknowledge by gaze. The speaker's presented signal—which might consist in the production of a turn—will be identified, perhaps by making use of a clarification question. The examples (6) and (7) provided in Fig. 2 both contain clarification questions related to phonetic form, cf. Nina's - ***have a car?*** as well as Morris' and June's exchange, repeated here in the abridged version (9):

(9) a. Morris: [$\cdots$] *a man called* ***Annegra?*** -

b. June: *yeah,* ***Allegra***

c. Morris: ***Allegra,*** *uh replied* [$\cdots$]

Presentation of a signal by a speaker regularly implies signalling that p and thereby meaning that p. The correspondence on the addressee's side is recognition that p and understanding that p. In talk exchange presentation of a signal usually amounts to production of an utterance[2]. The presentation of a signal can be reacted upon by the addressee in various ways. He may indicate understanding by asserting, paraphrasing, by verbatim repetition, by completing the speaker's utterance himself or by the initiation of a next turn.

Speaker's contributions can be associated with the proposal of a joint project which may be considered and eventually taken up by the addressee. This is shown in Jane's and Kate's exchange in example (8), where the joint projects involved both are information exchanges. The first proposed by Jane concerns the whereabouts of Miss Pink and the second one involves the identity of the caller Kate.

According to Clark, "An action ladder is a set of cotemporal actions ordered with upward causality, upward completion, and downward evidence. In language use these levels are joint actions" (1966, p. 154). It may well be the case that the leading metaphor "action ladder" is not too well chosen. However, the concept, besides showing broad coverage of linguistic data, allows for incrementality, partiality, default interpretation and bottom-up as well as top-down processes. Especially data like self-repairs and other-repairs support the Clarkian notion of layered structures.

Needless to say that matters are hardly ever clear cut: We have under-specification wrt to indication, identification, recognition and understanding, considering proposals and reporting on actions carried out. One of the most conspicuous features in our corpus is that the default interpretation of addressees' reply behaviour as successfully carried out action is a frequent source of error. It leads to extensive back-tracking and rebuilding operations.

Granted that our estimation of data is correct and that there is some truth in the action ladder concept, we better look for its merits. The notion of a joint project provides us with the property of strong cohesion and

[2]In order not to complicate matters too much, we exclude information on more than one channel here most of the time.

dependence we seem to need: Initiation of a joint project implies more, we can conclude, than producing the first element of an adjacency pair. It means to get the addressee to join in on a common purpose. The initiator of the project takes the first step towards reaching the proposed goal. If his proposal succeeds, it will be defined as the common goal throughout the transaction. At the same time a "we"-intention will be built up providing the ground for the agents' maintained belief that the task set is to be jointly achieved. Thus we can answer our question (b), "How are the agents' consecutive acts related to each other?", by stipulating that the necessary cohesion is provided by their attempt to set up a joint project[3] (see section 5 for an example and attempts at more detailed explanation).

As we will see in section 4, coordination with respect to single utterances will be one of the driving forces in the evolution of a dialogue. Ideas concerning question (b) are hence also relevant for the discussion of question (c), to which we now turn.

3.3 Discussion of question (c) "What are the principles on which the alternation of agents' acts is based?"

An answer concerning (c) must resort to the public information/common ground built up by the agents in the following way: The joint project envisaged must be such that a change of agent/role is called for. The change of agent/role must be licenced by the information present in the common ground. A short illustration of this point with respect to questions and directives might be of help: If a felicitously asked question exists, it can be treated as public and an answer can be given. If a directive was accomplished, a perlocutionary effect can be triggered and a reply can be produced indicating successful action. In both cases public information/common ground is changed. In the question case information has been provided, in the directive case the situation was altered.

We are now prepared to embark on a preliminary discussion of our example (5), repeated here for the reader's convenience as (10):

(10) a. A: *A man walks in the park.*

b. B: *He whistles.*

Concerning (10) we arrive at still new questions triggered by our discussion of their kindred (a), (b) and (c). They concern the joint project involved, the reason for the change of the speaker and the first speaker's consent, respectively:

(d) What is the joint project which A and B intend/try/want to pursue?

[3]Clarifying the relation of the "joint-project" concept to SDRT's rhetorical relations (see e.g. Lascarides and Asher (1999)) would be a worthwhile project to follow up upon. Roughly, the main difference seems to be that joint projects are action-and-agent-oriented, whereas rhetorical relations are structure-oriented, hence, dependent on the compositional semantics of dialogue contributions.

(e) Why did addressee B take over and appoint himself next speaker?
and, above all,

(f) Why did speaker A let addressee B take over and appoint himself next speaker?

Clearly, finding answers to (d), (e) and (f) would add to our knowledge why we can take (10) as an adjacency pair. So let's try. Given (10) and no other information whatsoever the questions are extremely difficult to answer. We could introduce a variant of Lascarides' and Asher's rhetorical function of narration (see Lascarides and Asher (1999)), distributed across speakers. But this, if feasible at all, falls short of explaining the change of the speaker role. We also know from data involving multi-agent narrations that these have prototypical rules of their own. Especially the introductory phase of multi-agent narration, turn-allocation and control of the information provided by the potential story-tellers is highly standardized. None of this is present in (10), however.

The attempt at explanation we propose is the following: Contributions in discourse may, in addition to their more obvious function of providing information, initiating action etc., provide clues concerning matters of successful coordination. B appointing himself next speaker may for example indicate high degree of coordination concerning the scene seen. This also would explain, why A does not react to the fact that the role of current speaker is snatched away from him. In chapter 6 we will see that this is consequential upon a deeper regularity.

4 Information in Dialogue Can Be on Diverse Things

Our assumptions concerning the structure of dialogue have been so far that joint projects employing adjacency pairs are the building blocks of dialogue. A cursory look into dialogue corpora quickly reveals, however, that dialogue structure is more complicated and more diverse. Complexity and diversity characterize the information flow as organized by the participants of the discourse. Concerning complexity and diversity various researchers, notably Schegloff (1979), McNeill (1992), Clark (1994, 1996) have observed that dialogues are roughly organized in

(a) material directly related to primary task/topic and

(b) negotiations related to emerging problems of various kinds.

Fig. 2 already provided evidence showing this type of organization. Clark has it that the (a)-information and the (b)-information travel on so-called "tracks" and I will stick to this terminology here. The track-metaphor suggests that we have different lines of information in dialogue which are interrelated but form separable units nevertheless. Problems can in due course give rise to new problems (see example (11)). Hence, tracks can be embedded recursively. They are opened up and closed according to regular procedures. In addition, tracks are running in parallel. This is perhaps most evident from situations where information from various channels (vision, speech) is used.

Example (11) shows a dialogue track structure, its embeddings as well as its openings and closings. We are in an airplane-construction task and the situation is as follows: The instructor tells the constructor to fasten part of the wing to the fuselage. (I provide a rough translation from the German transcript.)

(11) a. Instructor: <pause ..> *And there beneath an orange one* <pause ..>. ***Don't know it's name.***

b. Constructor:***Rhombus.***

c. Instructor: ***Uhm?***

d. Constructor:***Rhombus.***

e. Instructor: ***The orange rhombus*** <both laugh> *into the last* [hole].

f. Constructor:*O.K.*

The construction state is depicted in Fig. 4, a visualization of the track-structure of (11) is given in Fig. 5.

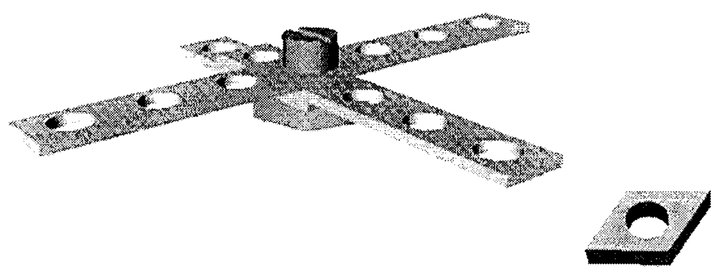


Figure 3: Wing construction and isolated nut fixing the bolt, called *rhombus* in (11), see also Fig. 5

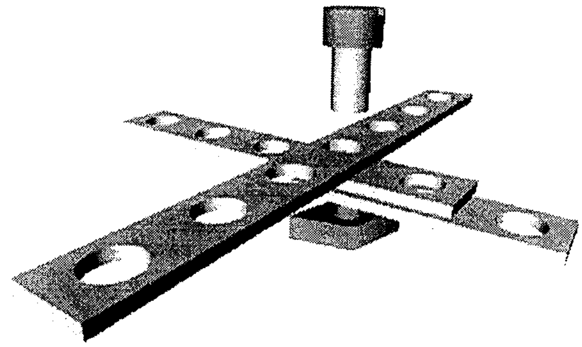


Figure 4: Construction state: Instructor tells constructor to fix part of the airplane's wing to the fuselage.

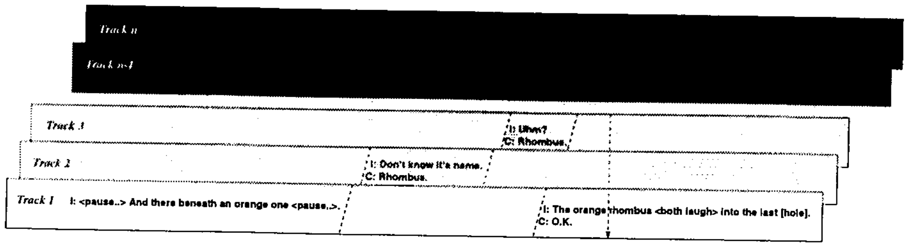


Figure 5: Visualization of the track-structure in example (11).

Turning to (11), we now want to gather a few observations concerning tracks. It is clear that a word-finding problem causes the instructor to interrupt his current directive. He can't complete it because he does not know the name of the object to be chosen by the constructor. Seen from the perspective of speech act theory, he lacks one of the fundamental preconditions for the production of a relevant instruction. Hence, before the construction process can continue, this problem has to be solved. It is made public by the instructor. Making it public and thus indicating non-satisfaction of a pragmatic presupposition of his directive moves the instructor from track 1 (where he should provide information but can't) to track 2, where wants to get his word-finding problem solved in order to continue on track 1 again.

The constructor cooperates, however his attempt first leads to a new problem. Expressed in terms of Clark's action ladder (see Fig. 2), the instructor cannot identify the signal and starts a clarification question. Since the existing signal on track 2 has not yet been decoded, the clarification question is generated on track 3. The constructor's answer on track 3 is accepted by the instructor. Since track 3 has been collateral to track 2, the track 2-activity can also be regarded as completed. So work on track 1, which has been kept open and "running" all the time, can be resumed again. The instructor produces a repair using a definite description *the orange rhombus* acting as an anaphora to the original *reparandum* which is *an orange one*.

Two things should be noted here in passing, which are not central to our present concern, although they shed some light on the role of coordination in non-regimented task-oriented dialogue. They deal with the production of the anaphora and the identification of tracks in (11). Observe that the use of anaphora depends on a successful joint project, initiated by the instructor and completed by his uptake of *rhombus*. Strictly speaking, there is no grammatical antecedent for the definite NP *the orange rhombus*. In order to resolve its anaphorical link, we have to resort to the agents' coordination process and to the information it generates. Seen this way, information on higher tracks serves as a precondition for the generation of information on track 1. Concerning tracks in (11) we have an embedding procedure to depth three and then a sort of stepwise closing down of recursive moves. However, in particular the step down from track 2 to track 1 is not clearly marked. The invented example (12) below was designed to show more clearly what happens in terms of Clark's "action ladders", its track construction is presented in Fig. 6.

(12)	Instructor:	<pause ..> *And there beneath an orange one* <pause..>.	
		Don' t know it's name.	
	Constructor:	***Rhombus.***	
	Instructor:		***Uhm?***
	Constructor:		***Rhombus.***
	Instructor:		***Mhm* (identification of word), *rhombus* (understanding).**
	Instructor:	***The orange rhombus*** <both laugh> *into the last*[hole].	
	Constructor:	*O.K.*	

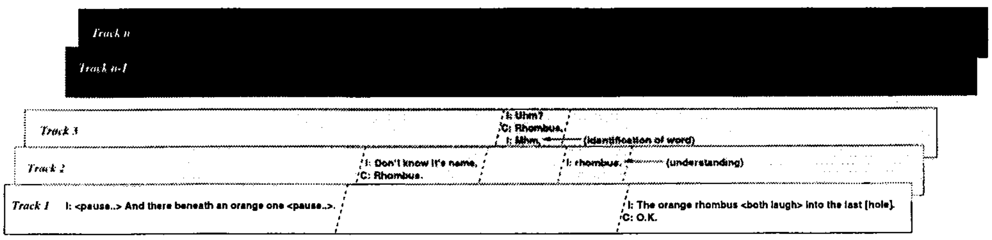


Figure 6: Track construction with respect to invented example (12)

The "upwards embedding" of example (12) is as in example (11). However, now the difference comes: On track 3 the instructor signals that he could identify the word and that he also got its semantics. The rest is as before.

A word concerning the constructor's *O.K.* in (11) should be in order since its use is quite characteristic for the sort of underspecification going on in dialogue. As it stands, we do not know whether *O.K.* is meant to indicate one of the following:

- understanding of the semantic and pragmatic impact of the definite description
  or

- identification of the orange rhombus
  or

- signalling that the joint project of identifying the nut has been accomplished
  or

- demonstrating the intention to take over the instructor's intentions at this construction stage
  or

- replying that the action has been carried out.

We have now brought together a good many of observations and can start to discuss what examples (11) and (12) can teach us concerning dialogue.

Evaluation of Examples (11) and (12)

Examples (11) and (12) give us a hint concerning the structure of natural discourse. Discourse structure can best be viewed as an ensemble of parallel activities systematically interlinked. This tructure is called "track"-structure here in order to do justice to H. Clark who seems to have been the first researcher to give it a central place in his theorizing. But nothing whatsoever hinges on the terminology chosen. Perhaps "opening up channels" or "side sequences" would do as well to convey the idea[4]. Whatever ones pet terminology, the following facts remain:

- joint projects are established on different tracks and—as a consequence—adjacency pairs, speech acts etc are used in different functions, mostly in order to secure preconditions,

- the change of tracks is frequently indicated, although the range of signals used as indicators is not well known, especially the means of discourse phonology applied (lengthening of vowels etc.),

- "upwards embedding" is more clearly marked than "downwards closing" and coming back to track 1.

Clearly, someone working without using the track idea would also have to take account of these findings. If taken seriously, the items listed would have to affect our understanding and our theories of dialogue in the following way: The structure of turns would have to contain branch points for embedding tracks. As a consequence, the ideas concerning micro- and macro-structures in dialogue would have to change. Macro-structures could no longer be based on speech acts only. Somehow the idea of joint projects would have to be accounted for. It must be admitted though that much depends on results in research on signalling the opening and closing of tracks. A side

[4]Jan van Kuppevelt and Joris Hulstijn argued during the *Amstelog*-conference that working with a system of side sequences would do here as well. This remains to be investigated in the context of empirical research concerning regularities of opening up and closing tracks.

effect of taking the intricate information structure in dialogue seriously is that timing problems become important: A new track has to be opened at the relevant stage in discourse. Also, activity on it cannot go on for ever. It has to be brought to an end in due time.

At times I've seen it written down that natural language discourse contains object language and meta-language at the same time, although I cannot attribute that to a particular philosopher—is it Quine?, Wittgenstein II?, Strawson?, Lewis? or Kamp? Anyway, the final point I want to make in this context is that higher order tracks can contain information related to semantic and pragmatic interpretation. One can grasp this fact in the following way: Higher order tracks contain the information necessary to interpret expressions and actions on lower tracks, in other words, they provide a partial characterization of the immediately relevant model of interpretation. This holds, above all, for settings in which there is no face-to-face-contact among the agents. I try to show this feature with respect to the somewhat demanding case of metonymy interpretation.

Consider example (13a). The situation in which the relevant speech data are generated is as follows: The constructor does not know how to locate an aggregate which will turn out to be part of the airplane's tail (see Fig. 7 below). At the time of the instructor's intervention, however, the constructor does not know that it is the tail that is under construction at present. Here the instructor comes in with his contribution:

(13) a. Instructor: *Look, the whole thing is going to be an airplane. The five-carrier you just fixed* <pause .> ***that is the, that is the tail unit, the elevator unit of the airplane, that is to say the rear part.***

b. Constructor: *Yes.*

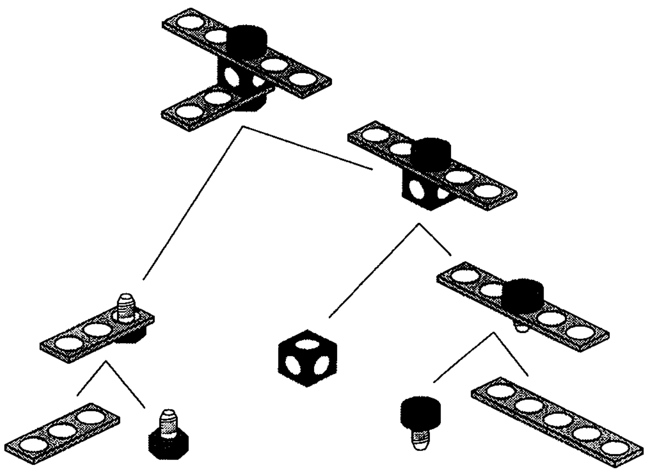


Figure 7: Airplane's tail

Suffice it to note here that the instructor's remark provides the information for a successful and speedy completion of the task.

By now, everyone should be convinced that there is something to this idea of tracks structuring dialogues. Concerning tracks, many problems remain, of course, which should not be brushed aside here. One problem concerns the existence of tracks and the other their being marked in one way or other. Do one-track dialogues exist at all? That this is the case seems to be an implicit assumption of most existing approaches to dialogue. However, we know from face-to-face-exchange that non-verbal information is always present, providing, for example, information about understanding and acknowledging. It is still far from clear, however, how we can distinguish tracks from each other, if we do not have explicit verbal signals as in example (11).

The answers we can provide to these "tracky" questions will shed light on the problem of what can be considered THE structure of (task-oriented) dialogue. By the way, I do not think that we have as yet arrived at a satisfactory understanding of dialogue structure.

We now turn to our next topic, the introduction of new objects into an already existing scene.

5 Introduction of New Objects in Task-oriented Dialogue

In order to discuss the topic of object introduction in task-oriented dialogue, we introduce example (14) below. It is taken from an airplane setting where the agents were screened off from each other. As a consequence, public information/common ground has to be established by purely verbal means. At first sight, this might be considered a rather artificial situation, leading to non-natural data. It turns out, howver, that the situation in face-to-face settings is quite similar: Here the he agents have different positions with respect to the objects to be manipulated, hence their perspectives on objects may differ. In order to overcome problems with aggregates or atomic parts agents will rotate them, open up clarifying side sequences etc. Using this setting and the data produced from it, I want to show that agents provide evidence for what can count as public information concerning the objects introduced. That is my major point here.

The sub-dialogue (14) is translated from German and was slightly regimented. The situation is as follows: The instructor tells the constructor that his task is to build an airplane. This is expressed by *It will be an airplane.* Afterwards he introduces the parts the constructor will need to set up part of the fuselage as well as the connection between the fuselage and the tail. The five-holes bar is needed for the fuselage, the three-holes bar must be used for the link. Having introduced the objects needed, the instructor describes the type of construction to be worked on, i.e. the overlap of the three-holes bar and the five-holes bar as well as the overlapping bars' positions, see fig. (8).

(14) a. Instructor:*It will be an airplane.* [...] *You first take* $[_{\text{indef.NP}}$ ***a***

one, two, three, four a five <pause ..> ***five-holes bar with five holes in it***] <pause ..>.

b. Constructor: *Yes.*

c. Instructor:*And* [$_{\text{indef.NP}}$ ***another one with*** <*pause* ..>**three holes**].

d. Constructor: *Mhm.*

e. Instructor:*Then you place* [$_{\text{def.NP}}$ ***the one with the three holes***] *beneath* [$_{\text{def.NP}}$ ***the one with the five holes***] *and let two holes overlap.* <pause .>. *O.K.?*

f. Constructor: *O.K.*

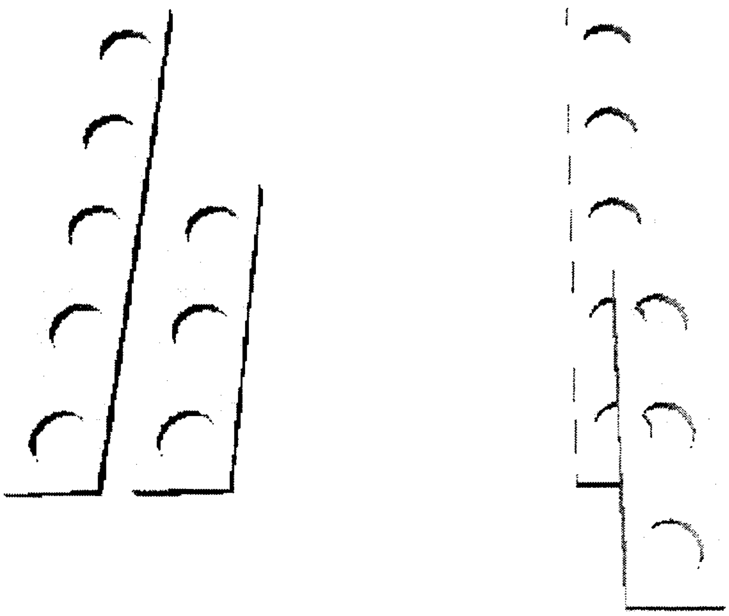


Figure 8: The overlapping bars as described by example (14)

Looking at the sub-dialogue (14), we notice the following things:

- The instructor carries out a counting action in order to produce a felicitous object description. The counting action interferes with the production of the expression. Most probably, this should be considered a track 2 phenomenon. We will not discuss this in the sequel, however.

- We have the instructor's responses *Yes, Mhm, O.K.*

- The instructor's last directive is followed by a question *O.K.?* asking whether the constructor reacted to the directive. Here the cautious formulation *reacted* is used on purpose. It is not clear, what the intended topic of the question is. See the following comment on the instructor's response below.

Using Clark's action ladder recipe we can say that the constructor's responses *Yes, Mhm, O.K.* can be related in principle to understanding, intention, commitment to task or to task carried out, in other words, the constructor's responses are underspecified. Metaphysically speaking, one could expect a clarification question from the instructor in these cases. But nothing happens of this sort and coordination is somehow going on: Agents form beliefs about what is the case on the other side. The constructor will form a belief that there is a construction involving the two bars on the instructor's side. The instructor can take the constructor's *Yes* and *Mhm* as a signal that the latter picked up the bars.

Two things deserve some note here: The instructor's belief is based on a default and should be regarded as defeasible, provided the underspecification argument given above is correct. The constructor's belief about his situation comes from visual information, which is perhaps safer. What can emerge now is public information based on belief. The belief's content is that there are two bars introduced on the constructor's side and two parallel bars involved in a construction on the instructor's side. The public information attained can be regarded as the entity, usually called "common ground", in our case based on mutual belief.

The common ground in turn provides information on two things: First, the precondition for the directive are specified and the antecedents for the two definite NPs are set up. If the instructor could not rely on the fact that the bars were introduced on the constructor's side, he could not refer back to them by anaphora. Since he has the constructor's words, he can rely on the antecedents' being fixed. Now, one could argue that the setting is too special to allow for far-reaching generalizations. But remember that I argued above that perspective and overlap of objects in face-to-face situations can be compared to the effects of the screen.

Summing up we can say that the use of definite anaphora in dialogue is based on the agents' coordination and on the emerging public information. Agents have to make sure that an object has been introduced by opening up a new track and starting a side sequence. Cooperativity seems to dictate that they provide some evidence by verbal or gestural means.

6 Constructive Deconstruction: Re-situating the Walker-in-the-Park

The dialogue examples from the airplane corpus seem to differ so much from our introductory example (5) that it seems to be difficult to learn something from them for the latter's description, especially for the treatment of anaphora. In order to see the relation between (5) and the more natural (6) to (8), we try to convert (5) into a life-like dialogue. The first step is to invest (5) with situatedness. So, we have to find a situation into which (5) could be embedded. Here it is:

Imagine that A and B are guards which prevent the public from invading some park. Some day, standing on the watch-tower, they perceive a Donnellanesque shape moving across the lawn in the not-too-far distance. A might then say *A* man_{A}? *walks in the park* using a so-called "try-marker" (indicated by "?") and the dialogue might then continue as delineated in example (15).

(15) a. A: *A* man_{A}? *walks in the park.*

b. B: *Yeah, and* he_{B} [$_{\mathrm{f}}$ *whistles*].

c. A: *Right,* he_{A} *whistles.*

Using the insights we arrived at when commenting upon Clark's "action ladders" and "tracks" etc., we can now venture the following hypotheses: A suggests a joint project to B which we can name "classification of situation and securing evidence". Such joint projects are e.g. to be found in didactic contexts like "master–apprentice" situations or in less hierarchically structured environs where evaluating action alternatives depends on a close investigation of the properties of the larger situation one is in (as may happen, say, in mountain climbing where one has to look for appropriate handholds).

The type of situation needed for (15) is one where a change of the speaker role arises in a natural manner. By the way, one reason, why question–answer pairs in dialogues are preferred by researchers to theorise on is that there the data, so to speak, already solve this question for you: joint project and change of speaker role are clear from the outset (see Ginzburg (1999)). However, if we have contributions distributed among different agents matters are less clear.

Maintaining the fiction we established, we can say that A describes to B the incident. Hence, we are on track 1 at the beginning. However, since the observed shape is of a Donnellanesque kind, A does not fully trust his own classification. Therefore he opens up track 2 by a so-called try-marker "?", intimating that he suggests a classification of the object in the park, which, however, could as well perhaps be a small dynosaurus in a rain-coat. A's description of the object is marked as being open to negotiation. He indicated that by his "?" and thereby implied that he is embarking on a joint project. He suggests at the same time that A's and B's common aim should be to solve the classification problem before the further description of the scene can go on.

B agrees to the common aim. He takes up A's proposal for classification by his *yeah*, but it remains to be decided in which way. *Yeah* may be either related to the introduction of the object or to the classification of the whole situation, i.e. to the proposition expressed by A's contribution. Example (16), involving a correction, shows that *yeah* could be related to object introduction only.

(16) a. A: *A* man_{A}? *walks in the park.*

b. B: *Yeah, but* he_{B} [$_{\mathrm{f}}$*hovers over the ground*].

Furthermore, B could agree to A's classification of the object-in-the-park's property but contradict A's object classification. This is the classical Donnellan case, cf. (17):

(17) a. A:*A* man_A? *walks in the park.*

b. B:*No, it is a* [$_f$*woman*$_B$] *and she* [$_f$ *whistles*].

c. A:*Right, she*$_A$ *whistles.*

Note, by the way, that (18), showing that A agrees but hangs on to his original decision concerning gender matters would be weird or have to get a very special interpretation:

(18) a. A:*A* man_A? *walks in the park.*

b. B:*No, it is a* [$_f$*woman*$_B$] *and she* [$_f$ *whistles*].

c. A:*Right, he*$_A$ *whistles.*

After this small digression probing into the scope of *yeah*, we come back to (15a) and (15b). Clearly, both classifications of A's should be considered defeasible. At this stage of the dialogue A and B can believe it is public information that the object out there in the park is a man and walks.

Note the difference of (15a) to our original example (5): Track 2 was opened by A and has to be closed in order to let the dialogue proceed (at least this is the default option, there are others, however, see below). This explains B's *yeah*. Now B can base his anaphora on the public information and this determines the grammatical form of the pronoun. Hence, the following (19b) could not be considered well-formed.

(19) a. A:*A* man_A? *walks in the park.*

b. B:*Yeah, and she*$_B$ [$_f$ *whistles*].

Furthermore, indeterminacy concerning negotiation on track 2 can lead to marked anaphoric NPs (20b):

(20) a. A:*A* man_A? *walks in the park.*

b. B:*Yeah, and* <pause> *your man*$_B$ <pause>[$_f$ *whistles*].

Here the *yeah* seems to confirm classification of the object's walking property. Being a man, however, is, so to speak, put into scare quotes. Anyway, in (15b) B adds that the man whistles. He marks his information by contrastive accent. So, whistling said of the man is the new or rhematic information entered into the dialogue. A agrees by *right* etc. to the new information. Should *right* be considered as information residing on track 2? Should A's total supplied information be attributed to track2? I am not sure here. Whatever the correct answer might be, A's use of the pronoun is licenced, since felicity of antecedent formulation was in no way questioned by B. Observe the oddity of (21c):

(21) a. A:*A man*$_A$? *walks in the park.*

b. B:*Yeah, and he*$_B$ [$_f$ *whistles*].

c. A:*Right, she*$_A$ *whistles.*

After these explorations in the acceptability of dialogues, we take up the case of (15c) again: A agrees to the whistling. Hence it is public information among A and B concerning a man out there that he walks and whistles.

What is at the heart of cross speaker anaphora in cases like this? A and B must share a belief (i.e. it is public information) that there is an object out there which bears such-and-such properties . They must also have evidence that they share the relevant belief. In (15) we can "read that off" the dialogue. Here A and B are obviously guided by strong cooperativity principles.

7 Conclusion

Dynamic semantics has paved the way for a successful treatment of anaphoric relations in non-dialogical text. Here DRT, SDRT, DPL and other variants of Dynamic Semantics (like Chierchia's 1995 book), however, the solutions proposed seem to resist a straight-forward generalisation to multi-speaker dialogue. In addition, dialogues are more complex than the usual A-says-and-then-B-says-examples suggest. Dialogues can be viewed as joint projects serving a common purpose. Even if one does not want to subscribe to Clark's track idea one must admit that dialogues contain a main line of information designed to further its principal aim. Interwoven with the main line we have subsidiary lines marked in various ways, e.g. phonologically, lexically, by turn-constructional units or by gesture, given visibility plays a role. The use of anaphora for objects introduced seems to be intimately tied up with what can be considered as information residing in the common ground. This can serve as an explanation for the fact that agents invest quite some time and energy to make clear what they consider to be in the common ground or, what amounts to the same here, what they believe to be mutual belief. This seems to indicate that anaphora resolution must rest on agents' emerging beliefs.

If my observations and arguments are plausible, then the situation in describing talk exchange is fairly different from the one standardly treated in DRT and DPL. Nevertheless, what one ultimately wants, are, of course, solutions similar to those found in the Dynamic Semantics tradition. A promising start in an interesting direction was made in H. Kamp's (1990) paper but I am not aware of any larger project following this line of research the descriptively minded linguist could use to handle his dialogue data. Observe that Kamp's paper also seems to imply that if a speaker uses anaphora in non-dialogical text, he himself has to believe that he talks about an entity introduced by an antecedent. Well, this hooks up to my remark at the beginning that dialogues should serve as a prototype for theorizing about non-dialogical texts and not the other way round.

References

Anderson, C. A. and Owens, J. (eds.): 1990, *Propositional Attitudes: The Role of Content in Logic, Language and Mind*, CSLI

Asher, N.: 1993, *Reference to Abstract Objects in Discourse*, Kluwer Academic Publishers

Asher, N.: 1998, *Common Ground, Correction and Coordination.*, The University of Texas at Austin

Benz, A. and Jäger, G. (eds.): 1997, MUNDIAL'97—*Munich Workshop on Formal Semantics and Pragmatics of Dialogue*, CIS, Univ. Munich

Chierchia, G.: 1995, *Dynamics of Meaning—Anaphora, Presupposition, and the Theory of Grammar*, Chicago UP

Clark, A. G.: 1994, Beliefs and Desires Incorporated, *Journ. Phil.* 91(8), 404–25

Clark, H. H.: 1996, *Using Language*, Cambridge UP

Clermont, T., Meier, C., M., P., Prestin, E., Rieser, H., Ritter, H., and Velichkovsky, B.: 1995, *Augenbewegung, Fokus und Referenz*, Technical Report 95/8, SFB 360

Corpus, *Wir bauen also jetzt ein Flugzeug. Konstruieren im Dialog, Arbeitsmaterialien. Interaktion sprachlicher und visueller Informationsverarbeitung*, Collaborative Research Center SFB 360 "Situated Artificial Communicators", Bielefeld University

Dekker, P.: 1997, *On First Order Information Exchange*, in: Benz and Jäger (1997)

Dekker, P. and van Rooy, R.: 1998, *Intentional Identity and Information Exchange*, MS for ITALLC '98

Ginzburg, J.: 1998, *Clarifying Utterances*, in: Hulstijn and Nijholt (1998)

Ginzburg, J.: 1999, Clarification in dialogue: Meaning, content and compositionality, in *Amstelog99 [Preproceedings]*

Givon, T. (ed.): 1979, *Syntax and Semantics*, Vol. 12, Academic Press

Groenendijk, J. and Stokhof, M.: 1991, Dynamic Predicate Logic, *Linguistics and Philosophy* 14, 39–100

Heydrich, W., Kühnlein, P., and Rieser, H.: 1998, *A DRT-Style Modelling of Agents' Mental States in Discourse*, in: Hulstijn and Nijholt (1998)

Heydrich, W. and Rieser, H.: 1995, *Public Information and Mutual Error*, Technical Report 95/11, Universität Bielefeld, SFB360

Heydrich, W. and Rieser, H. (eds.) in cooperation with Peter Kühnlein: 1998, *Mutual Knowledge, Common Ground and Public Information*, European Summer School in Logic, Language and Information, Universität des Saarlandes Saarbrücken in cooperation with DFKI Saarbrücken

Hulstijn, J. and Nijholt, A. (eds.): 1998, *Formal Semantics and Pragmatics of Dialogue—Proceedings Twente Workshop on Language Technology 13 (Twendial '98)*, TWLT, Universiteit Twente, Faculteit Informatica

Kamp, H.: 1990, *Prolegomena to a Structural Account of Belief and Other Attitudes*, in: Anderson and Owens (1990)

Kamp, H. and Reyle, U.: 1993, *From Discourse to Logic—Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*, Vol. 42 of *Studies in Linguistics and Philosophy*, Kluwer

Kawamori, M. and Shimazu, A.: 1999, Speech acts and dialogue control in task-oriented spontaneous dialogue, in *Amstelog99 [Preproceedings]*

Lascarides, A. and Asher, N.: 1993, Temporal interpretation, discourse relations, and common sense entailment, *Linguistics and Philosophy* 16, 437–93

Lascarides, A. and Asher, N.: 1999, Cognitive states, discourse structure and the content of dialogue, in *Amstelog99 [Preproceedings]*

McNeill, D.: 1992, *Hand and Mind—What Gestures Reveal about Thought*, Chicago UP

Poesio, M.: 1998, *Cross-speaker Anaphora and Dialogue Acts*, in: Heydrich and Rieser (1998)

Poesio, M. and Traum, D.: 1998, *Towards an axiomatisation of dialogue acts*, in: Hulstijn and Nijholt (1998)

Pomplun, M., Prestin, E., and Rieser, H.: 1998, *Eye-Movement Research and Dialogue Structure*, Technical report, Universität Bielefeld, SFB 360

Schegloff, E. A.: 1979, *The relevance of repair to syntax-for-conversation*, in: Givon (1979)

van Rooy, R.: 1999, *Attitudes and Changing Contexts*, MS, University of Amsterdam (draft)

Permission to change*

Robert van Rooy

ILLC/University of Amsterdam
the Netherlands
vanrooy@philo.uva.nl

Abstract

In this paper I discuss how to account for the performative effects of imperatives, and concentrate mainly on permission sentences. In the first part of the paper I argue that the performative effects of permission sentences should be accounted for in terms of a context change theory by making use of *contraction*. In the second part the perhaps main problem for this analysis is discussed, i.e. the problem of conjunctive permission sentences. I develop two ways to solve this problem. First, I argue that perhaps this problem is due to the wrong way of accounting for contraction, and I propose several alternative ways in which contraction can be defined that account for the performative effects of conjunctive permissions in a more satisfiable way. Second, I show how we could also account for the problem by means of a type-shift analyis.

1 Introduction

According to Austin's speech act analysis, by using sentences we make certain linguistic acts. The traditional problem for speech act analysis was to find interesting types of speech acts, and to find necessary and sufficient conditions for the successful performance of the act. In later theories of context change the emphasis was on the essential *effects* of the speech acts made. In Stalnaker's (1978) classical analysis of assertions, for instance, the essential effect of an assertion is said to be the change it brings about in what is commonly assumed; the context that represents what is commonly assumed is incremented with the content of the assertion. More recently,

*This paper was written while I was working on the 'Sources and Streams of Information'-project, sponsored by the Dutch Organisation for Scientific Research (NWO). I would like to thank Frank Veltman and especially Ede Zimmermann for discussion.

the essential effects of other speech acts, like questions and denials, have been analysed in terms of context change, too. The goal of this paper is to follow up some classical papers of Kamp and Lewis to characterise in a context change theory the essential effects of *imperatives*, normally stated in terms of command and permission sentences of the form *You must/may do A*.

In sections 2 and 3 I will discuss whether we can account for the performative effects of command/obligation and permission sentences by treating them assertorically, or whether we should treat them as explicit performatives. I will argue for the latter option. This gives rise to the question *how* command and permission sentences change the context. In the fourth section I will show how we can account for the change when we represent the prior information state by a preference order, and how in terms of this change governed by this preference order we can predict most intuitions about coördination connectives and quantificational determinors. In section 5 I discuss the problem how imperatives determine the preference order of the posterior information state, and relate the account with Veltman's analysis of defaults in update semantics. In the sections that follow it I will concentrate on a particular problem for our analysis that was noted first by Merin (1992); the problem of *conjunctive* permissions. First I will show how the standard analysis of contraction used for the analysis of permission sentences gives rise to this problem, and in the last three sections I will suggest three ways in which the problem can be solved when we give up the assumption that all that counts are the propositions that are expressed by the embedded clauses of permission sentences.

2 Performative effect in terms of assertion

According to Austin's classical analysis of speech acts, sentences of the form *You must/may do A* are normally not used to describe a states of affair; they are rather used to *do* things. In terms of the language game between master and slave as described by Lewis (1970/9), they are typically used by one person, the master, to command or permit another person, the slave, to do certain things.

How should we account for these so-called *performative* effects of the sentences used by the master? One proposal might be to say that command and permission sentences are *assertorically* used, but that the performative effect is accounted for in an *indirect* way, due to the fact that we *learn*, or realise, more about the world (Kamp, 1979). One of the things one might learn about the world is what is demanded and permitted. A truth conditional analysis of what is demanded and permitted is given in deontic logic. Standard deontic logic (SDL) was based on the same principles as classical modal logic.[1] Where normal modal logic has the oper-

[1]There are other truth-conditional analysis of deontic concepts, of course. The analyses of Hansson (1990) and Meyer (1993) certainly belong to the most interesting ones. Other approaches

ators $\Box$ and $\Diamond$ standing for respectively necessity and possibility, SDL has the two operators O and P. They stand respectively for *ought* or *obliged* and for *permission.* Just like $\Box$ and $\Diamond$ are duals of each other, also P is defined in terms of O as follows: $P(A) \equiv \neg O(\neg A)$ for any proposition A. Model theoretically it is said that $O(A)$ is true in w iff in all ideal worlds accessible from w, A is true, and $P(A)$ is true iff A is consistent with this set of all ideal worlds.[2] The set of ideal worlds in w will be denoted by $P(w)$, and is known as the permissibility set. According to the standard theories of context change for assertions, the effect of the successful assertion that A is the case is that the context, K, which represents what is commonly assumed by the participants of the conversation, changes from K to $\{w \in K |\ w$ makes A true$\}$.[3] Note that according to the truth conditional analysis used in SDL, what the agent is obliged and permitted to do is a *fact* about the world. What is permitted in different worlds might be different; i.e. $P(w)$ need not be the same as $P(w')$ for any two worlds w and w' in K. If we now learn that John is permitted to make A true, we learn something about the world. As in standard context change theory this can be accounted for by eliminating those worlds w from K where $P(w)$ contains no A-worlds. We might now propose that the performative effect of command and permission sentences is due to the fact that only after a command or permission sentence is used by one person, the slave *knows* that he is obliged/permitted to do something, and acts accordingly.

This assertoric analysis seems appropriate for certain uses of permission sentences. But there are certain problems with the analysis, too. First, a truth-conditional analysis of imperatives gives rise to the expectation that command and permission sentences can be embedded into one another, which in fact seems impossible.[4] Second, it is rather questionable whether the performative effect of all permission sentences should be accounted for in the *epistemic* way sketched above. Consider the following sentences:

(1) a. You may take an apple,

b. You may take a pear, and

---

claim that to talk about deontic discourse, we should also take the notion of *tense* seriously. Although these approaches all have something to say, I won't discuss them here, but I should note that they go some (but not the whole) way in solving the problems we will discuss in this paper. For a discussion of a number of approaches towards the closely related analysis of boulatic concepts, see Van Rooy (1999).

[2] In this paper I will use capitals for both sentential clauses and the propositions expressed by them. I hope this will never lead to confusion.

[3] If we forget about introspection, and consider only monotone updates.

[4] I should note, though, that also the standard truth conditional analysis can account for the intuition that the formula $O(O(A))$ should mean the same as $O(A)$; just assume that the accessibility relation is both transitive and dense. A deontic accessibility relation is dense if every deontic alternative is a deontic alternative to a deontic alternative.

c. You may take an apple or take a pear.

According to SDL, (1c) follows from both (1a) and (1b), and neither (1a) nor (1b) follows from (1c). Normally, however, if we make a disjunctive permission, both disjuncts are also permitted. The most straightforward way to account for this strong reading of disjunctive permissions would be to build it into the formal system by adding an axiom which has that result. Unfortunately, if we assume that $P(A \vee B) \Rightarrow P(A) \wedge P(B)$ is a theorem of the logic, we can derive in SDL that everything is allowed. Let us abbreviate $P(A \vee B) \rightarrow P(A) \wedge P(B)$ by FCP, for free choice permission. Because in normal deontic logic we can derive $O(A \vee B)$ from $O(A)$, and $P(A \vee B)$ follows by definition of P from $O(A \vee B)$, the assumption that FCP is a theorem of deontic logic has the unwanted consequence that from $O(A)$ we can derive $P(B)$ for any A and B; everything is permitted. For instance, suppose you are allowed to walk in the park, $P(A)$. By logic, this means that you are also allowed to walk in the park or kill the king, $P(A \vee B)$. If PCP were valid, this would mean that allowing you to walk in the park, also allows you to kill the king.

Thus, already for logical reasons alone, our analysis should not obey disjunction elimination. But the assumption of disjunction elimination is empirically wrong, too. As Kamp (1979) observed, there is nothing problematic with the assertion of the following permission sentence:

(2) You may take an apple or a pear, but I don't know which.

We can conclude that we should not make FCP valid by stipulation. An assertoric analysis of permission sentences based on standard deontic logic indeed does not allow for disjunction elimination. It remains mysterious, however, how such an analysis can account for the fact that *normally* we conclude from a permission of a disjunction to the permission of its disjuncts.[5]

There is another phenomenon that the SDL-truth conditional analysis cannot explain. This is the fact that quantificational determinors should be read differently in permission sentences than in obligation/command sentences. Note that permission sentences obey existential weakening, i.e. we are allowed to infer (3b) from (3a):

(3) a. You may eat three apples.

b. You may eat an apple.

Indeed, this is predicted by the standard truth-conditional account, because permissions are predicted to be closed under logical implication. But what this account

[5]But see Kamp (1979) for some proposals by using Gricean conversational implicatures, and Merin (1992) for a critical discussion.

cannot predict is why quantifiers under the scope of *may* get the 'at most' reading, while in the scope of *must* they get the 'at least' reading. Intuitively, after (3a) is said by the master John is allowed to eat none, one, two, or three apples, but not more. But if the master says that you *must* eat three apples, it is still possible that you may take more than three apples. On the standard truth-conditional account, however, this cannot be explained, because quantifiers always get the same 'at least' interpretation.

3 The performative analysis of imperatives

The natural alternative to the assertoric analysis of obligation and permission sentences is the *performative* one. According to the performative analyses of Lewis (1970/9), Stalnaker (ms), and Kamp (1973), command and permission sentences are not primarily used to make true assertions about the world, but rather to *change* the world, by changing what the slave is obliged/permitted to do in that world.[6] With some feeling for Amsterdam rhetorics we might say that according to the performative analysis, we know the meaning of an imperative sentence, when we know how imperatives change permissibility sets.[7]

According to this Lewis/Kamp/Stalnaker account, if the master commands John in w to do A by saying *You must do A*, or allows John to do A in w by saying *You may do A*, it is typically not yet the case in w that the proposition expressed by A is respectively a superset of, or consistent with, John's permissibility set, $P(w)$. However, the performative effect of the command/permission will be such that in the new world, w', the command and permission sentences will turn out to be true. Thus, in case the command or permission is not used vacuously, $P(w')$ will be different from $P(w)$, such that with respect to w' the obligation/permission sentence will be true.

Note that according to this analysis the performative effects of command and permission sentences are very similar to the performative effect of declarative sentences that are assertively used, when we take seriously the Stalnakerian (1978) assumption that what is commonly assumed (presupposed) is a propositional attitude; a propositional attitude that the agents have in the actual world. Thus, we should think of what our context is as a fact of our world; what we commonly assume in our world, w, might be different from what is commonly assumed in the counterfactual world v; $K(w)$ need not be the same as $K(v)$. Because the attitudes agents have is a fact about the world, when the information that is commonly assumed

[6] Although Lewis (1970/9) and Kamp (1973) account for the effect of permission sentences in rather different ways, both might be called performative analyses in the sense that their effect is to change the permissibility set.

[7] cf. Tan & Van der Torre (1997).

changes because something new is accepted, this can only mean that the world has changed. Thus, the context change through the succesful assertion of A should not be modeled by the change from K to $\{v \in K | \; v$ makes A true$\}$, but rather by the change from w to w', where $K(w')$ is very much like $K(w)$ except that all worlds in $K(w')$ also make A true.[8] Notice that just like the obligation *You must do A* can be true in w although A itself is not true in w, it can also be the case that through an earlier assertion of A, A is now commonly assumed in w, although A itself is not actually true in w. The analogy between assertions and imperatives extends also to their non-recursiveness; it simply doesn't make sense to embed one speech act into another.

Although the performative effects of succesful assertions and of succesful commands/permissions have, according to the above suggestions, more in common than you might have expected, there still exists also an important difference between the two. According to the above analysis of assertions, we are in a particular world, but we don't know in which one. Moreover, assuming that the attitude of 'commonly assumed' is introspective,[9] it is quite irrelevant to know in which world we are. This is rather different from what we have to assume in our above analysis of commands and permissions; there it is very relevant that we know what the actual world is, because it determines the permissibility set which the master and slave have to know in order to understand the performative effect.

But what if we don't know what the actual world, w, is, i.e. if we think it might be any world in $K(w)$? Then we still might go for a performative analysis of command and permission sentences if we assume that the permissibility sets of all worlds in $K(w)$ are very similar to one another. What is permissible is modeled as a fact about the world. Let us say that v and u are P-indistinguishable, $v \approx_P u$ iff $P(v) = P(u)$.[10] Now we can assume that even if we don't know in which world we are, we can still go for a performative analysis if for any two worlds v and u in $K(w)$ it holds that $v \approx_P u$. This can be the case, for instance, because only a few explicit commands had been given that determine the permissibility set. But in fact we don't have to make this strong assumption. All we have to assume is that in the initial context, $K(w)$, all permission sets are inconsistent with A when a permission sentence of the form *You may do A* is used performatively. What happens if a command or permission is given? What will happen is that the world changes from w to w', where $K(w')$ will be different from $K(w)$ because in the worlds of $K(w')$ the permission sets are different from the worlds in $K(w)$. Let us assume that $f_v(May(A))$ is the (unique) world that is the 'same' as v except that its permissibility set is the 'minimally' different permissibility set such that it verifies

[8] If we forget about introspection.

[9] An information state $IS(w)$ is introspective iff for each $v \in IS(w)$ it holds that $IS(v) = IS(w)$.

[10] Notice that the notion of P-indistinguishability gives rise to an accessibility function, PI, defined as $PI = \lambda v.\{u \in W | \; P(u) = P(v)\}$, which partitions the set of worlds.

the truth conditional content of $May(A)$.[11] Then we can define the change from $K(w)$ to $K(w')$ due to the performative use of a sentence of the form *You may do A* as follows: $K(w') \stackrel{def}{=} \{f_v(May(A)) | \; v \in K(w)\}$.[12]

Although it is natural to assume that sentences of the form *You may/must do A* are normally not used in an assertoric way, it seems that they can be used in this way, as is made clear by the use of (2). In terms of our framework we can easily account for this assertoric use of command and permission sentences, and say how this use differs from the performative use. We might say that while in the performative use the permissibility set of each world changes through the use of the sentence, in the assertoric use the permissibility sets remain the same, and the worlds in $K(w)$ that do not make obligation or permission sentences true are simply eliminated. Although the effects of the two uses of permission sentences are accounted for in rather different ways, they have one thing in common; after the succesful context change, it will be the case that for every v in the new context $K(w')$, it holds that $P(v) \cap A$ is non-empty. We can conclude that although the primary effect of command and permission sentences is, according to the performative approach, to change the permissibility set, the approach is not (or need not be) inconsistent with a truth conditional analysis after all; in the new context $K(w')$ all worlds verify the truth conditional content of the permission sentence.

In the rest of this paper I will ignore the possibility that in different worlds the permissibility sets might be different. Instead, I will assume that in each context there will be only one permissibility set around. But if knowing the meaning of an imperative means that you have to know how the imperative changes the permissibility set, our main task is to say how command and permission sentences govern the change from the prior permissibility set, P, to the posterior one, P'.

4 Changing by contraction

For commands this problem seems to have an easy solution. If the command *You must do A* is given by the master, the new, or posterior, set of permissible futures for John, P', is intuitively simply $P \cap A$.[13] However, things are more complicated for permission sentences. It is clear that if A is allowed, P' should be a superset of P such that $P' \cap A \neq \emptyset$. It is not clear, however, which A-worlds should be added to P. Obviously, we cannot simply say that $P' = P \cup A$. By that suggestion, an allowance for A would allow everything compatible with A, which is certainly not

[11]In the next section we will see that the permissibility set of $f_v(May(A))$ will be denoted by $P(v)^-_{\neg A}$.

[12]This kind of context change is known as *imaging* and is rather different from the kind of context change we will discuss in section 4 below.

[13]What if the new command is incompatible with one or more of the earlier ones? In that case we might make use of change by revision to be discussed below.

what we want. But how then should the change from P to P' be determined if a permission is made? This is Lewis's problem about permissions.

Stalnaker (ms.) suggested that Lewis's problem about permissions can be solved when we know not only what the best worlds are, but also assume that some of the non-ideal worlds are better than others. Thus, to account for the performative effects of commands and permissions, we need not only a set of ideal worlds, but rather a whole preference, or reprehensibility, ordering, $\leq$, on the set of all possible worlds. On the interpretation that $u \leq v$ iff v is at least as reprehensible as u, it is natural to assume that this relation should be reflexive, transitive, and connected.[14] We might assume such a primitive preference relation among worlds, but we can also follow Harper (1976) and determine an ordering relation on worlds by looking at the number of obligatory propositions that worlds make true. Van Fraassen (1973) and Kratzer (1981) have argued that to account for moral deliberation we should not rule out the possibility that the command-sentences that determine what should be done might be mutually inconsistent. Let O be the set of propositions that John is obligatory to make true. Then we say that u is at least as desirable as v with respect to O, $u \leq_O v$, iff the set of commands in O that v makes true, $|\{A \in O|\ v \in A\}|$, is smaller than the set of commands in O that u makes true, $|\{A \in O|\ u \in A\}|$, where $|S|$ is the cardinality of S.[15] In terms of this preference order on possible worlds we can determine the set of worlds that make as much as possible commands true; the minimal elements of the relation $\leq_O$. Thus, if we call this set of minimal elements P, P can be defined as follows:

$$P \stackrel{def}{=} \{v \in W|\ \forall u:\ v \leq_O u\}^{16}$$

In terms of this set of ideal worlds we can, as before, determine of course whether according to the present state A is obliged or permitted.[17]

But this ordering relation contains more information than just what the set P of ideal worlds is, and in terms of this extra information we can determine the new permissibility set P'. If the master permits the slave to make A true, we can assume that P contains no A-worlds, i.e. none of the A-worlds is ideal. But some A-worlds are still better than other A-worlds. Stalnaker proposes that the effect of allowing A

[14] A relation R is *reflexive* if for all w : $R(w,w)$, it is *transitive* if for all w, v and u: if $R(w,v)$ and $R(v,u)$, then $R(w,u)$, and it is *connected* if for all w and v, $R(w,v)$ or $R(v,w)$ or $w = v$.

[15] Van Fraassen (1973) and Kratzer (1981) use a different way to determine the preference relation. They say that $u \leq_O v$ iff $\{A \in O|\ v \in A\} \subseteq \{A \in O|\ u \in A\}$. In this way, $\leq_O$ determines a partial ordering, but not a total one. Not all worlds have to be connected with each other.

[16] Note that if the propositions in O are mutually consistent, P is just $\bigcap O$.

[17] For instance, A is obliged iff $P \subseteq A$. Notice that in terms of our preference order we also might go for various other analyses of obligation. Just to mention one, we can say that the slave is obliged to make A true iff $\forall \langle v, w\rangle \in \leq$: if $w \in A$, then $v \in A$. Notice that this analysis of obligation, in distinction with the standard one, does not predict that obligations are closed under logical implication.

is that the best A-worlds are added to the old permissibility set to figure as the new permissibility set. In modern terminology (cf. Gärdenfors (1988)), we might say that the change form P to P' due to a permission sentence is accounted for in terms of *contraction*, the rational retracting of information from an earlier information state, where contraction is governed by a reprehensibility ordering $\leq_O$, which I will denote by $\leq$ from now on. I will define contraction in terms of revision, and the revision of P by any proposition A can be defined in terms of the relation $\leq$ as follows:

$$P^*_A \stackrel{\text{def}}{=} \{u \in A |\ \forall v \in A : u \leq v\}$$

If we define for any set of worlds K, $v \in K^*_{\{v,w\}}$ iff $v \leq w$, it turns out that our revision function '$*$' will satisfy the following constraints on minimal change (cf. Harper (1975)):

(K^*1) For any proposition A, $K^*_A \subseteq A$

(K^*2) If $A \neq \emptyset$, then $K^*_A \neq \emptyset$

(K^*3) If $K \cap A \neq \emptyset$, then $K^*_A = K \cap A$

(K^*4) If $K^*_A \cap B \neq \emptyset$, then $K^*_{A \wedge B} = K^*_A \cap B$

The first condition demands that the new state should contain the new information. The second requires that changing with new consistent information results in a new consistent information state. The third and fourth condition demand that the change is conservative; if A is consistent with K, K^*_A is simply K added with A, and if B is consistent with K^*_A, then receiving the information B after the information that A has the same effect as receiving the information A and B together. These rules are the analogues of the familiar AGM-postulates for revision,[18] when it is assumed that information states and propositions are modeled by sets of possible worlds. If we now define the *contraction* of K by A, K^-_A via the Harper identity as $K \cup K^*_{\neg A}$, it follows that also the usual AGM-postulates for contraction will be satisfied.[19]

To implement Stalnaker's suggestion, we can say that the change induced by the permission *You may do A* is that the new permission set, P', results from P by

[18] Alchourón, Gärdenfors and Makinson (1985). See Gärdenfors (1988) for a general overview of theories of belief revision.

[19] By representing information states by sets of possible worlds, the AGM-postulates for contraction are the following:
(K^-1) $K \subseteq K^-_A$, (K^-2) $K \not\subseteq A \Rightarrow K^-_A = K$, ($K^-3$) $A \neq \top \Rightarrow K^-_A \not\subseteq A$, ($K^-4$) $K \subseteq A \Rightarrow K^-_A \cap A \subseteq K$, ($K^-5$) $K^-_{A \wedge B} \subseteq K^-_A \cup K^-_B$, ($K^-6$) $K^-_{A \wedge B} \not\subseteq A \Rightarrow K^-_A \subseteq K^-_{A \wedge B}$.

contracting $\neg A$: $P' = P^-_{\neg A}$, which is $P \cup P^*_A$.[20] Thus, according to this proposal, command and permission sentences change a context of interpretation as follows (where P' is $Upd(OP(A), P)$):

$$\begin{aligned} Upd(Must(j, A), P) &= P \cap A \\ Upd(May(j, A), P) &= P^-_{\neg A} = P \cup P^*_A \end{aligned}$$

Note first that if change by permission is governed by the reprehensibility ordering, the 'at most' interpretation of the quantificational determinors used in the embedded sentences is immediately explained. If according to the prior state John was not even allowed to take one single apple, one might assume that worlds where he takes only one apple are 'closer' to the worlds in P, than worlds where he takes more. So, after the permission that he may take an apple, the new permission set, P', can be expected to contain only worlds where he takes at most one apple.[21] Note that by our analysis it is also predicted that the quantificational determinors in command sentences get the usual 'at least' reading, just like desired. Thus, our performative analysis of command and permission sentences can predict and explain certain intuitions about the behaviour of quantificational determinors that the standard truth conditional, or assertoric, analysis cannot.

Note that according to our performative account it does not follow that for a permission sentence of the form *You may do A or B* the slave can infer that according to the new permissibility set he is allowed to do any of the disjuncts, nor is the arbitrarily interpretation of indefinites guaranteed. Still, the performative analysis can give an explanation why indefinites and disjuncts are normally interpreted in this 'free-choice' way. To do this, let me first define a deontic preference relation between propositions in terms of our reprehensibility relation between worlds, $\leq$. We can say that although both A and B are incompatible with the set of ideal worlds, A is still deontically preferred to $B, A \preceq B$ iff the best A-worlds are closer to the ideal worlds than the best B-worlds, $\exists v \in A$ and $\forall u \in B: v \leq u$.[22] Then we can say that with respect to $\leq$, A and B are equally strong reprehensible, $A \approx B$, iff $A \preceq B$ and $B \preceq A$. Because we defined contraction via the Harper identity in terms of revision, and because any revision function that obeys $(K^*1) - (K^*4)$ satisfies the following factoring condition: $K^*_{A \vee B} = K^*_A$ if $A \prec B$,[23] $K^*_{A \vee B} = K^*_B$ if $B \prec A$, and $K^*_{A \vee B} = K^*_A \cup K^*_B$ if $A \approx B$, we can now explain why normally disjunction elimination is allowed for permission sentences. For simple disjunctive permission

[20] This analysis of permission sentences, and thus of contraction, was in fact assumed by Kamp (1979) and Merin (1992) in their discussions of the performative analysis of permissions.

[21] Rohrbauch (1996) wrongly claimed that the possible world framework as such always predicts the *at least* readings for quantified permission sentences like (3a) and (3b).

[22] The epistemic variant of this ordering relation between propositions is called the epistemic *entrenchment* relation.

[23] Where $A \prec B$ iff $A \preceq B$, but not $B \preceq A$.

sentences like *You may do A or B*, it is not unreasonable to assume, I think, that by Gricean, or strategic, reasoning we can conclude (perhaps after accommodation) that the master has normally no strict preference for the one above the other.[24]

This doesn't mean that this reasoning can be accounted for in a straightforward way. It would be nice to explain the strong reading completely in terms of *conversational implicatures.* Kamp (1979) shows, however, that by the way conversational implicatures are normally understood, as inferences that take as one of their arguments the proposition expressed by sentences, these implicatures can be of no help to explain the strong reading of disjunctive permissions. The problem is that these strong readings should also be predicted in case disjunctive permissions are embedded in larger sentences such that the proposition expressed by this larger sentence does not entail the proposition expressed by the embedded disjunctive permission sentence. The following example is given:

(4) Usually you may only take an apple. So if you may take an apple or take a pear, you should bloody well be pleased.

To account for the strong reading of performatively used disjunctive permissions it seems that we have to build the implicature into the meaning of *or*, that is, that the relevant implicature is not a conversational one, but a *conventional* one instead. But then, how should we account for this conventional implicature? We might propose that disjunctions of the form $P \vee Q$ can only be appropriately interpreted in contexts K such that $P \cap \neg Q \cap K^*_{P \vee Q} \neq \emptyset$ and $\neg P \cap Q \cap K^*_{P \vee Q} \neq \emptyset$.[25] As a result, it is predicted that a normal sentence of the form A *or* B can only be appropriately asserted in a context K that is compatible with both $A \wedge \neg B$ and $\neg A \wedge B$,[26] and that a permission sentence like *You may do A or B* can only be said appropriately by the master to John iff he is *indifferent* between A and B, $A \approx B$.

Kamp (1973) argued that just like we can define an inference relation between propositions, we might also define an inference relation between performatively used permission sentences, which he called 'p-entailment'. In terms of our framework, he proposes that permission sentence $May(A)$ p-entails permission sentence $May(B)$, iff for every appropriate initial permissibility ordering $\leq$, no new worlds would be added to the set of ideal worlds through the use of $May(B)$ after the initial permissibility set was 'updated' through the use of $May(A)$. On the assumption that

[24]But this doesn't always seem to be the case; as noted by Kamp (1973), even for the performative analysis of permission sentences the arbitrarily interpretation of at least indefinites can be cancelled: *You may pick a flower, but don't pick a rose.* Kamp also notes that we can force this arbitrarily interpretation by using the negative polarity item *any.*

[25]Zimmermann (1999) has recently argued that this condition does not count as an appropriateness condition for disjunction, but is part of the *semantic* meaning.

[26]This is exactly what Stalnaker (1975) proposes to be the acceptability conditions for disjunctive sentences.

$\leq$ can only be an appropriate initial permissibility ordering for the performatively used $May(A \vee B)$ iff $A \approx B$, we can conclude that both (1a) and (1b) are p-entailed by (1c).[27] Some readers might see that Kamp's notion of p-entailment is rather close to Veltman's (1996) fixed point notion of entailment between speech acts.[28]

5 Changing preferences

Lewis (1979) complained that an account in terms of reprehensibility-gradings of worlds might handle single cases of permissions, but leaves undetermined how the comparative permissibility relation evolves from permission to permission.

Actually, the problem exists already for the case of commands. But here the problem can be solved almost straightforwardly in a way made familiar to us by the analysis of defaults in update semantics by Veltman (1996). According to one version of Veltman's system, an information state, $\langle K, \leq \rangle$, consists of two parts; (i) a set of possible worlds, K, which represents the factual knowledge we assume, and (ii) our reflexive and transitive ordering relation, $\leq$, on worlds. Until now we have assumed that for the analysis of the command *You must do A* we only need to look at the best worlds according to $\leq$, and we only gave a partial description of the resulting state; we only defined what the *best* worlds are in the new information state, not what the whole new ordering relation will be. But given Veltman's analysis of 'normally', it is quite straightforward to say how the new ordering relation after the update with the command should look like:

$$Upd(Must(A), \langle K, \leq \rangle) \quad = \quad \langle K, \{\langle v, w \rangle \in \leq \mid \text{if } w \in A, \text{ then } v \in A\} \rangle^{29}$$

Veltman's update semantics can be used to account for the performative effect of normal commands or obligations, but it also seems to be the natural framework to account for *conditional obligations* like *If Jesse robbed the bank, he ought to confess.* In the standard analyses of Hansson (1969), Lewis (1973), Spohn (1975), and Kratzer (1978), conditional obligations are treated solely in terms of our ordering relation $\leq$. Just like a standard obligation to do A is satisfied in the traditional truth conditional analysis when the best worlds according to $\leq$ are all A-worlds, they say that an obligation to do A on the condition that C is satisfied when the best C-worlds according to the ordering relation are all A-worlds. Notice that the analyses of conditional obligations by the above authors are all static; they only say when

[27]Notice that according to this entailment relation we don't predict that we can infer $May(A \vee B)$ from $May(A)$, and thus that Ross' paradox does not arise.

[28]See also Stalnaker (1975) for his notion of *reasonable inference*.

[29]This use of Veltman's update semantics for the analysis of commands was actually proposed by Tan & Van der Torre (1997).

a conditional obligation is true, not what the performative effect is of the use of a conditional obligation sentence.

Fortunately, this performative effect can be easily accounted for in terms of the framework Veltman (1996) uses for the analysis of *rules of exception.* To account for rules of exception in update semantics, Veltman no longer represents an information state by a pair like $\langle K, \leq\rangle$, where $\leq$ orders the worlds in W (and thus in K), but rather by a pair like $\langle K, \pi\rangle$, where π is a function taking as argument a proposition, A, and has as value an ordering relation $\leq_A$ restricted to the A-worlds. On the assumption that conditional obligations state rules of exception, the natural thing to say is that the information state $\langle K, \pi\rangle$ accepts the conditional obligation 'Must(B), if A' iff the best worlds according to the ordering relation $\pi(A)$ are all B-worlds. Now we can also account for the performative effect of conditional obligations as follows:

$$\begin{aligned} Upd((if A, Must(B)), \langle K, \pi\rangle) \quad &= \quad \langle K, \pi'\rangle, \text{ where } \pi' \text{ is such that} \\ \text{(i)} \quad & \forall C \neq A : \pi'(C) = \pi(C), \text{ and} \\ \text{(ii)} \quad & \pi'(A) = \{\langle v, w\rangle \in \pi(A) | \text{ if } w \in B, \text{ then } v \in B\} \end{aligned}$$

Although this analysis of conditional obligations is somewhat different from the traditional Hansson/Lewis/Spohn/Kratzer one, it shares with it an unwanted consequence. Note that both according to the traditional approach, and according to the analysis above, it is predicted that conditional obligations of the form *If A, then Must(A)* should be tautologically true. But that doesn't seem to be the case; the conditional obligation *If you rob banks, you should rob banks* is not trivially acceptable. Hansson (1969) already noted this prediction, but argued (somewhat mysteriously) that it is not really problematic. Spohn (1975) was not so confident about this anymore, and, in a more recent discussion, Frank (1997) argued that this prediction shows that the traditional analysis of conditional obligations must be on the wrong track.

If the traditional analysis of conditional obligations must be rejected because of the predicted validity of sentences of the form *If A, then Must(A)*, our analysis of conditional obligations must be rejected, too. But in distinction with the traditional approach towards conditional obligations, we can easily change the exact analysis such that the problematic prediction disappears. Note that in the above analysis we have assumed, just like Veltman (1996) did for the analysis of default rules, that the ordering relation $\pi(A)$ only orders the A-worlds. This has the result that the best worlds in $\pi(A)$ are by necessity all A-worlds. But, as we just saw, this is exactly what gives rise to the unwanted prediction that *If A, then Must(A)* must always be accepted. This suggests that our problem can simply be solved if we give up the assumption that for any A, $\pi(A)$ only orders the worlds in A; we simply assume that for all A, $\pi(A)$ orders the set of all worlds, W.[30] This doesn't mean that thus the condition has no effect anymore; it still might be the case that for different propositions A and B, the best worlds according to $\pi(A)$ are different from the best worlds according to $\pi(B)$. Then we might say that for the analysis of some conditional obligations we should only consider worlds that satisfy the antecedent, while for others we should not restrict the ordering relation in such a way.

We have seen that in terms of Veltman's update semantics we can determine how not only the set of best worlds evolve through the making of a (conditional) command, but even how the (set of) whole ordering relation(s) change(s). Although this is a stimulating result, it is still not clear how the ordering relation should evolve through the use of a permission sentence. The reason behind it is that the above analysis of the performative effects of (conditional) obligation sentences is essentially *eliminative*; through these sentences certain elements of the ordering relation that help to represent the information state are thrown away. The problem with permission sentences is that it doesn't seem to be possible to account for their performative effect in such an eliminative way. It follows that we need an account that determines how the ordering relation looks like in the posterior information state, even if the change cannot be accounted for by eliminating possibilities.

Notice that the problem we face is similar to the problem how to account for *iterated* belief *revision*. According to the standard analyses of belief revision, a prior information state is represented by an ordering relation like our $\leq$. If we revise this state with A, it is said that the posterior information state will be the set of best A-worlds according to $\leq$. But notice the difference in kind between prior and posterior information states; prior information states are ordering relations, while posterior information states are just sets of worlds. Because of this difference, the traditional approaches of revision can only account for so-called 'one-shot' revisions, but leave undetermined how iterated revisions should be accounted for. Meanwhile, however, there are several accounts of iterated revision around that can represent a posterior information state as an ordering relation, too.[31] Using any of those approaches towards iterated revision, we could analyse permission sentences as functions from one comparative permissibility relation to another. Because these analyses of iterated revision are technically somewhat more involved, and not of prime importance for the discussion here, I will ignore them, however.

[30]Notice that in this way we treat the conditional connective and the obligation operator in *If A, then Must(B)* separately. Something like this was first proposed, as far as I know, by Mott (1973), and defended recently also by Frank (1997).

[31]The first one, as far as I know, was developed by Harper (1976), while the approach of Spohn (1987) is, for good reasons, the best known.

6 Conjunctive permission sentences

In the discussion of Kamp (1979) the main concern was to account for the problematic disjunctive permission sentences. The above discussed analysis in terms of contraction goes a long way in solving this problem. But as noticed by Merin (1992), this analysis also gives rise to a new problem; the problem how to account for the performative effect of the use of *conjunctive* permission sentences. In the original Stalnaker account it is predicted that a conjunctive permission sentence *You may take an apple and a pear* has semantically the package deal effect: take either none or both. The reason is that $P^*_{A\wedge B}$ is a subset of $A \wedge B$, and thus that the only worlds in $P^-_{\neg(A\wedge B)}$ where either A or B are true, are worlds where *both* are true, if $P \cap (A \vee B) = \emptyset$.[32] But this package deal effect is empirically wrong, since a conjunctive permission allows also for the conjuncts to be done separately. Merin (1992) concludes that we should thus give up the analysis of permission sentences in terms of reprehensibility orderings, and that permission sentences stand in the way of Boolean and other lattice-theoretic semantics. In the rest of this paper I would rather like to take up the challenge to improve on the Stalnaker analysis, without giving up the assumption that the performative effect of permissions should be accounted for in terms of contraction.[33] Can we get rid of the package deal effect, and still analyse the effects of permissions in terms of contraction? And can we do this without even giving up a Boolean, or latice-theoretic, analysis of what is denoted by the embedded clause?

On first thought there seems to be an easy solution, based on a very simple intuition. Normally if the master allows you to do something, you can count on it that it is the 'worst' thing for him that he allows you to do. This means that when he allows you to do A, he also allows you anything that is less worse for him; i.e. he implicitly also allows you to make true any proposition B such that $B \preceq A$. This suggests that we should analyse permission sentences in the following way:

$$Upd(May(j, A), P) \quad = \quad \bigcup\{P^*_B \mid B \preceq A\}^{34}$$

[32] To a great extend, the problem is due to the validity of the *recovery*-postulate for standard contraction. Let me explain: By defining contraction in terms of revision by means of the Harper identity, it follows that if $K \subseteq A$, $K^-_A \cap A = K$, i.e. after contracting K with A, you can *recover* the original state K by learning A. In particular, this means that $P^-_{\neg(A\wedge B)} \cap \neg(A \wedge B) = P$. For the analysis of conjunctive permission sentences, standard contraction does not add enough worlds to the prior information state, because it seems that we want it to be the case that $P^-_{\neg(A\wedge B)} \cap \neg(A \wedge B)$ is a proper superset of P.

[33] What I will give up, however, is that the only thing that counts is the propositional content determined by the embedded clause of the permission sentence. In this sense I agree with Gazdar (1981) and others that for different speech act types, different types of denotations might be involved. Remember that according to almost any modern analysis of questions, for instance, the denotations involved are not simply propositions.

Notice for any propositions A and B such that $A \subseteq B$ it will be the case that $B \preceq A$. Because both A and B follow from $A \wedge B$, it is predicted that if it is allowed that $A \wedge B$, also A and B alone are allowed, which is exactly what we wanted.

Here is an example that suggests that the analysis must be wrong: His mother allows John to give a party while she is gone and clean the house afterwards. Allowing John to party and clean, $A \wedge B$, entails according to our analysis allowing John to party, A. But, then, his mother does not allow John simply to party, but only if he cleans afterwards. Fortunately, our analysis immediately accounts for this apparent problem. If his mother allows John to party, only the 'best' party-worlds are added to the set according to the reprehensibility ordering. It is natural to assume that in the given circumstances the best party-worlds (according to his mother) are only worlds where John gives a party *and* cleans the house afterwords. Thus, no world is added where he gives a party and doesn't clean, which shows that the apparent problem is not a real one.

However, the above suggested solution must be given up for another reason. It wrongly predicts that after the allowance of $A \wedge B$, not only A and B are allowed, but also everything else that is less reprehensible than $A \wedge B$, such as $(A \wedge B) \vee C$. Thus, if a proposition C that is not allowed in the prior context and is totally unrelated to A, but less reprehensible than A, the above interpretation rule would have the unwanted effect that the allowance of A also allows the agent to do C.

We have seen that by analysing permission sentences in terms of the standard contraction function, *not enough* worlds are added to the permissibility set after the permission to do $A \wedge B$, while analysing permission sentences in terms of our second contraction function would have the result that *too much* worlds are added. If we want to analyse permission sentences in terms of contraction, it seems that we need to define contraction such that the result of contracting P by $\neg A$ is somewhere in between $P \cup P^*_A$ and $\bigcup\{P^*_B |\ B \preceq A\}$.[35] What seems to be missing is a notion of *relevance*, i.e. if we are allowed to do A, not all worlds are allowed that are at least as good as the best A-worlds, but only the B-worlds than are at least as good as the best A-worlds for any proposition B that is *relevantly* entailed by A.[36] Thus,

[34]Because for any A, it will always be the case that $\top \preceq A$, and that for any P, $P^*_\top = P$, we don't have to define the resulting state in terms of the revised state united with the original state anymore.

[35]Notice that if we would define the contraction of P by $\neg A$ by $\bigcup\{P^*_B |\ B \preceq A\}$, this contraction function satisfies all AGM-postulates for contraction, except for (K^-4), the one responsible for recovery. From this observation, and the fact that the contraction function that we will actually use for the analysis of permission sentences in this section will be 'stronger' than this contraction function, it follows that also our contraction function to be defined below will satisfy all these AGM-postulates for contraction.

[36]For a nice discussion of the need for 'relevance contraction', see Cantwell (1999). His proposed analysis for relevant contraction is, however, different from mine, and won't help with our problem to account for conjunctive permission sentences.

we should analyse permission sentences as follows:

$$Upd(May(j, A), P) \quad = \quad \bigcup\{P^-_{\neg B}|\ B \preceq A\ \&\ A \mapsto B\}$$

where '$\mapsto$' is some kind of relevant entailment relation. Notice that by our reasoning above it holds that when our relevant entailment relation is stronger than the classical entailment relation, the analysis comes down to the following:

$$Upd(May(j, A), P) \quad = \quad \bigcup\{P^-_{\neg B}|\ A \mapsto B\}$$[37]

Our problem now comes down to finding a suitable entailment-relation that does the job. We have seen already that the ordinary entailment-relation is too weak. Is there any stronger notion of entailment around that does our job? Given our informal talk about 'relevance', it seems that we should take a look at Anderson & Belnap's (1962) relevance logic. But that would be wrong, for their relevance logic, just like classical logic, predicts that for any propositions A and B it follows that A entails $A \vee B$. We have seen already that this has the unwanted result that the allowance of A might allow the slave also to do C, for a C that is totally unrelated with A.

In the next two sections of the paper I will look at some proposals to find suitable candidates for our searched inference relation '$\mapsto$', and will relate the proposals to each other.

7 Type-shift

For the first kind of solution we will make use of a type-shift operator from entities of the proposition-type, to entities of the type corresponding to set of propositions.[38,39] If A is a proposition, we let F_A be the principle filter generated by A: $\{C \subseteq W|\ A \subseteq C\}$. Then it will be the case, for instance, that $F_A \cap F_B = F_{A \vee B}$. Now we can say that for the inference relation needed to account for permission sentences we should not look primarily at the normal semantic value of embedded clause A, but at (the minimal element(s) of) the 'filter'-semantic value, A^F. For reasons that will become obvious in a minute, I will not use A^F, but A^Q instead. The latter notion will not

[37] Where $P^-_{\neg B}$ is $P \cup P^*_B$, as before. In the informal discussion in the main text, I didn't call this function the actual contraction function anymore, but rather meant with the contraction from P with $\neg A$ the whole set $\bigcup\{P^-_{\neg B}|\ A \mapsto B\}$. It is perhaps good to note that if '$\mapsto$' is stronger than entailment, it still holds that $\bigcup\{P^-_{\neg B}|\ A \mapsto B\} \cap A = P^*_A$. Thus, even if the contraction function doesn't satisfy the recovery postulate, the revision with A is still equal to the contraction of $\neg A$ plus the adding of A (known as the Levi-identity in the belief revision literature).

[38] This type-shift operator is similar to the shift operator Winter (1995) uses to account for certain phenomena concerning conjoined NP's.

[39] For another, more familiar, use of type-shift, see section 9.

simply be the filter generated by A, but will be defined recursively. Let $A^Q = F_A$ if A is atomic, then we can define $(A \wedge B)^Q \stackrel{def}{=} A^Q \cap B^Q$ and $(A \vee B)^Q \stackrel{def}{=} A^Q \cup B^Q$. Observe that $(A \wedge B)^Q$, for instance, is not the same as $\{C \subseteq W | \ (A \cap B) \subseteq C\}$; just like all generalised quantifiers need not be filters, also A^Q need not be a filter. Assuming that $min(X) = \{e \in X | \ \neg\exists e' \in X : e' \subset e\}$, it is easy to see that the *minimal* element of $(A \wedge B)^Q$ is $A \vee B$, and that $(A \vee B)^Q$ contains two minimal elements, A and B.[40]

In terms of these 'lifted' meanings of A and B, we can now define the following inference relation:[41]

$$A \hookrightarrow B \quad \text{iff} \quad \begin{array}{ll} \text{(i)} & \forall e \in A^Q : \exists e' \in B^Q : e' \subseteq e, \text{ (iff } A^Q \subseteq B^Q\text{)} \quad \text{and} \\ \text{(ii)} & \forall e' \in min(B^Q) : \exists e \in min(A^Q) : e' \subseteq e \end{array}$$

Notice that if B does not classically follow from A, clause (i) assures that it will also not be the case that $A \hookrightarrow B$. But our notion of entailment is stronger than classical entailment (for this positive fragment), because of clause (ii). Although clause (ii) guarantees that we can infer A from $A \wedge B$, it does not predict that for any B it will be the case that $A \hookrightarrow A \vee B$.

As we have suggested above, this seems to be exactly the kind of inference relation needed. If we assume that the effect of an allowance of A with respect to P is defined as $\bigcup\{P^-_{\neg B} | \ A \hookrightarrow B\}$, it follows that by allowing $A \wedge B$, also A and B are allowed separately, if they are at least as good as $A \wedge B$, but no extra proposition C is allowed that is unrelated with $A \wedge B$.

Notice that our new analysis only differs from the old Lewis/Stalnaker analysis with respect to *conjunctive* permission sentences. *Disjunctive* permission sentences are treated similarly as before, giving rise to the prediction that both disjuncts are allowed only in case they were equally strong reprehensible before. But isn't it always the case that unless explicitly stated otherwise, as in (2), we infer from the permission to do $A \vee B$ that both disjuncts are allowed? Although I am not at all convinced of this, it might well be the case. Our analysis does not predict this yet, but we can, if we want, simply re-analyse the effect of the permission to do A as follows: $\bigcup\{P^-_{\neg B} | \ \exists C \in min(A^Q) : C \hookrightarrow B\}$. Because both A and B are elements of $min((A \vee B)^Q)$, this revised analysis would have the effect that the new permission set contains both A and B-worlds, just as desired.

You might complain that our analysis has the effect that the difference between conjunctive and permission sentences disappears. But, then, we might account for the difference in terms of appropriateness conditions. We might make the assumption that to make a disjunctive permission to do A or B appropriately, it

[40]At least, if A and B are atomic. In general $min((A \wedge B)^Q) = min(\{e \cup e' | \ e \in A^Q \text{ and } e' \in B^Q\}) = \{e \cup e' | \ e \in min(A^Q) \text{ and } e' \in min(B^Q)\}$. Similarly, $min((A \vee B)^Q) = min(A^Q) \cup min(B^Q)$.

[41]Tim Fernando (personal communication) informed me that this ordering relation between sets is known as the *Egli-Milner* preorder in Computer Science.

should be the case that with respect to the initial permissibility set, P, it holds that $P_A^* \cap P_B^* = \emptyset$. In that case, the permission to do A or B gets the *exclusive* reading; you are allowed to make A true, you are allowed to make B true, but you are not allowed to make both true. This in distinction with *conjunctive* permissions; if you are allowed to make A and B true, you are not only allowed to make A and B true separately, but also together. According to Merin (1992) this is exactly how it should be.

8 Events

Until now we have assumed that propositions are the basic building blocks of meaning, and that 'and' and 'or' should always be treated as intersection and union, respectively. Others have been more radical, and proposed that we should look at different kinds of primitives. A motivation for us might be that it seems that permission sentences take mainly *eventive* clauses as complements that are denoted by *to*-infinitives (cf. Rohrbauch (1996) and Portner (1997)). Rohrbauch argued, for instance, that for the analysis of permission sentences we need something like situation semantics, because we need a notion similar to, although not identical with, Kratzer's (1989) notion of *lumping*. He does not, however, give a formal definition of such a notion, but I will show how this can be done.

Proponents of this more radical solution typically take situations, facts, or events as primitives. Let us therefore look at one of the earliest analyses of such entities; Van Fraassen's (1969) analysis of events or facts.

Let us assume that an event-type is a set of events, and that an event itself is a set (of urelements, say). We will say that each eventive clause will denote an event-type. The model determines, as always, which event-type is denoted by atomic eventive clauses, and we might, but need not, assume that atomic eventive clauses denote sets that contain only one set of urelements. When the two eventive clauses A and B denote I_A and I_B, respectively, the disjunctive eventive clause $A \vee B$ will denote $I_A \cup I_B$, while the conjunctive eventive clause $A \wedge B$ will denote $I_A \times I_B$. The set $I_A \times I_B$ is the *product* of I_A and I_B determined as follows: $I_A \times I_B = \{e \cup e' | \ e \in I_A \text{ and } e' \in I_B\}$.[42,43] It is remarkable that for both simple and complex clauses, I_A

[42]Note that in distinction with Kratzer's situation-theoretic approach, but like Barwise & Perry's, we have *conjunctive* situations/events, but no disjunctive ones.

[43]It is worth to observe that Van Fraassen's analysis of events is closely related with Peleg's (1987) analysis of actions in his Concurrent Dynamic Logic. In this logic, actions denote relations between worlds and sets of worlds. Assuming that action α denotes P_α and that action β denotes P_β, in his semantics the denotation of action $\alpha \vee \beta$, $P_{\alpha\vee\beta}$, is $P_\alpha \cup P_\beta$ as in ordinary dynamic logic, and the denotation of the concurrent action $\alpha \wedge \beta$, $P_{\alpha\wedge\beta}$, is defined as follows: $P_{\alpha\wedge\beta} \stackrel{def}{=} \{\langle w, e \cup e' \rangle | \ \langle w, e \rangle \in P_\alpha \text{ and } \langle w, e' \rangle \in P_\beta\}$. As far as these connectives are concerned, we can conclude that we could have given our analysis also in terms of Peleg's variant of dynamic logic.

is closely related to A^Q. In fact, we might think of I_A as being $min(A^Q)$. This is interesting because it shows that by means of this alternative analysis of conjunction we can define the meanings of complexes directly in terms of the *minimal* elements of the 'quantifiers', the other elements are simply irrelevant.

Our purpose of using eventives was to be able to define a consequence relation that is stronger than ordinary entailment. Indeed, our main task remains to define an ordering relation '$\rightsquigarrow$' such that it accounts for similar things as the '$\hookrightarrow$'-relation did in the previous section. According to Rohrbauch this definition of '$\rightsquigarrow$' should also be closely related to Krazter's notion of *lumping.*

Giving our observation that we can think of I_A as being $min(A^Q)$, it is quite straightforward how we could define our wanted consequence relation (where $Q(A) \stackrel{def}{=} \{e' |\ \exists e \in I_A :\ e \subseteq e'\}$):

$$A \rightsquigarrow B \quad \text{iff} \quad \begin{array}{ll} \text{(i)} & \forall e \in I_A :\ \exists e' \in I_B :\ e' \subseteq e, \text{ (iff } Q(A) \subseteq Q(B)) \quad \text{and} \\ \text{(ii)} & \forall e' \in I_B :\ \exists e \in I_A :\ e' \subseteq e \end{array}$$

Notice that just like for any A and B it is the case that although $A \wedge B \hookrightarrow A$ but not $A \hookrightarrow A \vee B$, it is also the case that for any A and B, $A \wedge B \rightsquigarrow A$, but not $A \rightsquigarrow A \vee B$. Now we can use this consequence-relation for the analysis of permission sentences; i.e. the new permission set after the allowance to do A will be $\bigcup\{P^-_{\neg B} |\ A \rightsquigarrow B\}$, just like you expected.

As mentioned above, also Rohrbauch (1996) used events to account for the effect of permission sentences. He argues, however, that the permissibility set, P^e, should not be represented by a set of worlds, but rather by the set of events that are allowed. The permission of A simply increases this set. The question is, *How?* He argues that not only the 'minimal' events that make A true should be added to the permissibility set after A is allowed, but also the 'minimal' events that make B true, whenever B is a *natural part* of A. He does not know how to account technically for this notion of 'natural part', but given the discussion above, the proposal seems obvious. The new permissibility set should be $P^e \cup \bigcup\{I_B |\ A \rightsquigarrow B\}$.

Notice that according to this implementation of Rohrbauch's suggestion, a permission to do $A \vee B$ will always also give rise to the permission to do A and to do B. This is different from the way we have used eventives to analyse the effect of permission sentences ourselves, but just as in the previous section we can, if we want, account for this intuition, too. Just say that the permissibility set changes after the allowance to do A from P to $\bigcup\{P^-_{\neg B} |\ \exists C \in I_A : C \rightsquigarrow B\}$.

We have suggested that the entailment relation '$\rightsquigarrow$' is close to Kratzer's notion of 'lumping'. You would like to see *how*, and also how our analysis relates to the Boolean analysis. For both we need an analysis of *negation*. Let us make the Russellian assumption that with each atomic sentence p we can associate the 'urelements' $\mathbf{p} = I^+(p)$ and $\overline{\mathbf{p}} = I^-(p)$, whose occurrence in a world would make p minimally true and false, respectively. Assuming that $I_w^+(p) = \{\{\mathbf{p}\}\}$, if $\mathbf{p}$ holds in w, and $\emptyset$ otherwise, and something similar for $I_w^-(p)$, we can define $I_w^+(\neg A) \stackrel{def}{=} I_w^-(A)$ and $I_w^-(\neg A) \stackrel{def}{=} I_w^+(A)$ in the way familiar from partial logic, $I_w^+(A \wedge B) \stackrel{def}{=} I_w^+(A) \times I_w^+(B)$ as before, and on the assumption that '$A \vee B$' is, as usual, an abbreviation for '$\neg(\neg A \wedge \neg B)$', we analyse also $I_w^+(A \vee B)$ as before, if we say that $I_w^-(A \wedge B) \stackrel{def}{=} I_w^-(A) \cup I_w^-(B)$.

The classical notion of 'proposition' can obviously be defined in terms of the events: $[[A]] = \{w \in W |\ \exists e \in I_w^+(A)\}$.[44] Propositions, of course, behave Boolean, but what is, perhaps, remarkable to see is that $[[A]] \cap [[B]] = \{w \in W |\ \exists e \in I_w^+(A \wedge B)\}$, and $[[A]] \cup [[B]] = \{w \in W |\ \exists e \in I_w^+(A \vee B)\}$. This is interesting because it shows that if we want to give a uniform meaning to the conjunctive coördinator *and* in sentential and eventive clauses, we don't have to assume that *and* denotes Boolean intersection; we simply don't calculate primarily at the Boolean level.

If we say that $Q_w^+(A) \stackrel{def}{=} \{e' |\ \exists e \in I_w^+(A) :\ e \subseteq e'\}$, the 'quantifier' generated by $I_w^+(A)$, we can define Kratzer's notion of *lumping* as follows: *A lumps B* in w iff $Q_w^+(A) \subseteq Q_w^+(B)$.

Now we can extend our analysis of '$\leadsto$' to the case where we also consider negations. This relation can now be defined as follows:

$$A \leadsto B \quad \text{iff} \quad \begin{array}{ll} \text{(i)} & \forall w :\ A \text{ lumps } B \text{ in } w, \quad \text{and} \\ \text{(ii)} & \forall w :\ \forall e' \in I_w^+(B) :\ \exists e \in I_w^+(A) :\ e' \subseteq e \end{array}$$

It is interesting to see that Van Fraassen (1969) has proved that condition (i) comes down to *tautological entailment* in the sense of Anderson & Belnap (1962).

9 Lifting

In the last two sections we have assumed that in order to account for conjunctive permission sentences, we have to give up the assumption that we could account for the performative effect of permissions in terms of ordinary contraction. In this final section I want to show, however, that we could save our analysis in terms ordinary contraction, if we make use of our best known type-shift: *lifting*. The type-shift operator of lifting is the operator from expressions of type τ to expressions of type $\langle\langle\tau, t\rangle, t\rangle$, for τ any type. This type-shift operator is used in Partee & Rooth (1983), for instance, to conjoin expressions of type τ, with expressions of type $\langle\langle\tau, t\rangle, t\rangle$. Following Montague (1973), Keenan & Faltz (1985) propose that proper names, just like other NP's, should be analysed as being of such a higher type from the very beginning. They notice that expressions of type $\langle\tau, t\rangle, t\rangle$, are special in the sense that all denotations of this type can be thought of as Boolean combinations of the

[44] At least, if for each world w and atomic sentence p, either $I_w^+(p) = \{\{\mathbf{p}\}\}$, or $I_w^-(p) = \{\{\overline{\mathbf{p}}\}\}$.

principle filters generated by the expressions of type τ. The principle filter generated by the name *John*, for instance, can be denoted by $\{C \subseteq \wp(D) | \{\underline{John}\} \in C\}$, where D is our set of urelements. Notice that on a Boolean analysis of conjunction as a result the sentence *John* **and** *Mary danced* is true if and only if John danced and Mary danced. That is, the sentence can be true although John and Mary didn't dance together, and $\{\underline{John}\} \cap \{\underline{Mary}\} = \emptyset$, i.e. it matters *when* to apply Boolean intersection; application at the higher type results in a 'wide-scope' reading of '*and*', but application at the lower type does not.[45] Keenan & Falz notice that also for *that*-clauses, *whether*-clauses, and *to*-infinitives, it seems to make a difference when to apply the relevant operator in case the clause is coördinated; the (a) and (b)-sentences below can have a different interpretation (for instance when John only wants his mother to leave in case Mary comes):

(5) a. John hopes that Mary comes and that his mother leaves.

b. John hopes that Mary comes and his mother leaves.

(6) a. John knows whether Fred left or whether Mary left.

b. John knows whether Fred or Mary left.

Keenan & Faltz propose to account for this difference between (5a) and (5b), for instance, by assuming that the *that*-clauses do not denote propositions of type $\langle s, t\rangle$ that function as arguments of what John hopes (or Mary believes), but rather that these clauses denote entities of type $\langle\langle\langle s, t\rangle, t\rangle, t\rangle$, i.e. sets of sets of propositions, that take as arguments that what is denoted by 'John hopes' (the set of propositions that John hopes), or 'Mary believes' (the set of propositions that mary believes).[46] If we now assume that the word '*that*' is a type-shifting operator that shifts a proposition, A, of type $\langle s, t\rangle$ into the principle filter generated by A, $\{C \subseteq \wp(W) | A \in C\}$, of type $\langle\langle\langle s, t\rangle, t\rangle, t\rangle$, it will make a difference whether we apply the Boolean operations before or after the type-shift. Applying the operation *before* the type-shift results in a *small-scope* reading of the operator, but when it is applied *after* the type shift (which is the case in the (a)-sentences), a *wide-scope* reading of the Boolean operator results. Keenan & Faltz furthermore assume that natural language users can be rather sloppy, and sometimes (dear to) use (5b), when they really mean (5a). Thus,

[45]Hendriks (1987) allows for free type-shifting at any level, and is in this way able to account for the scope ambiguities of a sentence like *Every man and woman walks* by applying the conjunction before (small scope) or after (wide scope) the (optional) type-shift of the nouns *man* and *woman.*

[46]Keenan & Faltz note that in this way they can also account for sentences like *That Fred failed surprised John and annoyed Bill* without making use of old-fashioned conjunction reduction.

due to our laziness, the (b)-sentences, but not the corresponding (a)-sentences, are in fact ambiguous.[47]

Suppose now that we make similar assumptions for the embedded clauses of sentences like *You may A and/or B.* That is, suppose that also in this case a type-shift occurs from expressions of type τ to expressions of type $\langle\langle\langle\tau, t\rangle, t\rangle$, and let us for simplicity assume that also in this case '*that*' is the relevant type-shift operator from propositions to sets of sets of propositions.[48] As a result it will be the case that also for permission sentences the Boolean operators can apply either before or after the type shift. If we now would give a standard truth-conditional analysis of permission sentences, this would immediately result in a small- and a wide-scope reading of the Boolean operator, just like in the case of (5a) - (5b).[49] However, we have argued above that to account for the performative effect, our analysis should not be truth-conditional, but should rather be performative, by changing the permissibility set. The question now is whether and how we can use the type-shift to account for the performative effect of conjunctive permission sentences.

I will propose to use the type-shift by making use of the standard minimality operator:

$$min(X) \stackrel{def}{=} \{e \in X | \neg\exists e' \in X : e' \subset e\},$$

and by defining the update of P with *May(j,A)*, where A is a *that*-clause, as follows:

$$Upd(May(j, A), P) \stackrel{def}{=} \bigcup\{P^{-}_{\neg B} | \exists C \in min(A) : B \in C\}$$

Suppose now that the embedded clause is conjunctive in nature. Then it can either be of the form *that* $A \wedge$ *that* B, or of the form *that* $(A \wedge B)$. In both cases the denotation of the (simple or complex) *that*-clause contains only one minimal element,[50] but the minimal elements they contain are different. In the first case the minimal element itself contains two propositions, A and B, but in the second case the minimal element contains only one proposition, $A \wedge B$. By our definition of the update-operator, this means for the second case that only worlds are added that make both A and B true; the package deal, just like we had before.[51] When

[47] But notice that when we adopt Hendriks' free type-shift analysis, thus that type-shift is possible even though not triggered by explicit expressions or type-mismatch, we don't have to assume that the ambiguity is due to our laziness of using the word *that*; ambiguity is simply almost always around when Boolean operators are used and type shift is possible.

[48] Although this latter assumption is not crucial for the type-shifting analysis.

[49] Notice that when a negation occurs in the embedded clause of a permission sentence, this negation has typically a 'wide-scope' reading, and it not easy to see how to account for this on the earlier discussed analyses.

[50] If A and B are atomic.

[51] Although conjunctive permission sentences do not always give rise to the package-deal effect, we have seen in section 6 that some conjunctive permissions do in fact have a package-deal reading.

conjunction was applied *after* the type-shift, however, it is due to our definition of the update function predicted that both A-worlds, and B-worlds are added to the permissibility set, because the minimal element of the denotation of the *that*-clause contains both the proposition A and the proposition B. As a result, the new permissibility set might contain possibly some $A \wedge \neg B$-worlds and some $\neg A \wedge B$-worlds, just like we wanted.

It is worth to observe that our above analysis has also an interesting consequence for *disjunctive* permission sentences. First, consider the case where the disjunction of the embedded clause has *small scope* with respect to the lifting operator; $May(that(A \vee B))$. In that case disjunctive permissions have the same reading as in the original Stalnaker approach; the best $A \vee B$-worlds are added to the permissibility set. But now suppose that the disjunction has *wide scope* with respect to the lifting operator; $May(that\ A\ \vee\ that\ B)$. Now it will be the case, due to our definition of the update by permission sentences, that $May(that\ A\ \vee\ that\ B)$ has the same effect on the permissibility set as its corresponding conjunctive $May(that\ A \wedge that\ B)$, if A and B are atomic. This means that in distinction with the corresponding conjunctive case, the application of disjunction after type-shift has not really a widest scope reading of disjunction as a result. I am not sure whether this is right or not, but note that if we really want disjunction to have widest scope, and stick to our performative analysis of permission sentences, we have to give up the assumption that there is only *one* permission set arround. We rather have to assume that the context gives us a *set*, Q, of permission sets, and that we update the elements of this set as follows:

$$Upd(May(j,A),Q) \quad \stackrel{def}{=} \quad \{\{P^{-}_{\neg B} |\ B \in C\} |\ C \in min(A)\ \&\ P \in Q\}^{52}$$

This has the result that when $min(A)$ contains only one element, each element of Q is updated in the same way as in the earlier case. However, when $min(A)$ contains more elements, in case of a disjunction, from each element P of Q two or more new permission sets will result, depending on with which element of $min(A)$ permission set P is updated.

10 Conclusion

In this paper I have argued for a performative analysis of command and permission sentences. I have shown that by the standard Lewis/Stalnaker/Kamp analysis of permissions in terms of contraction most intuitions concerning coördinate connectives and quantificational determinors can be explained. An important problem for this approach turned out to be the analysis of conjunctive permission sentences. I

[52] A similar update-definition can be given for obligations, too, obviously.

have argued that, and shown how, this problem can be solved, when we either give a somewhat different account of contraction, or when we make use of type shift.

References

[1] Alchourón, C. E., P. Gärdenfors and D. Makinson, (1985), "On the logic of theory change", *Journal of Symbolic Logic*, 50, pp. 510-530.

[2] Anderson, A. and N. Belnap (1962), "Tautological entailments", *Philosophical Studies*, 13, pp. 19-52.

[3] Cantwell, J. (1999), "Relevant contraction", In: H. Rott et. al., *Proceedings of DGNMR'99*, ILLC, Amsterdam.

[4] Fraassen, B. van (1969), "Facts and tautological entailments", *Journal of Philosophy*, 66, pp. 477-487.

[5] Fraassen, B. van (1973), "Values and heart's command", *Journal of Philosophy*, 70, pp. 5-19.

[6] Gazdar, G. (1981), "Speech act assignments", In: Joshi et. al. (eds.), *Elements of Discourse Understanding*, Cambridge, Cambridge University Press, pp. 64-83.

[7] Gärdenfors, P. (1988), *Knowledge in Flux. Modelling the Dynamcics of Epistemic States*, MIT Press, Cambridge.

[8] Hansson, B. (1969), "An analysis of some deontic logics", *Nous*, 3, pp. 373-398.

[9] Hansson, S.O. (1990), "Preference-based deontic logic", *Journal of Philosophical Logic*, 19, pp. 75-93.

[10] Harper, W.L. (1975), "Rational belief change, Popper functions, and counterfactuals", *Synthese*, 30, 221.

[11] Harper, W.L. (1976), "Ramsey test conditionals and iterated belief change", In: W.L. Harper and C.A. Hooker (eds.), *Foundations of Probability Theory, Statistical Inference, and Statistical Theories of Science*, Vol I, Reidel, Dordrecht.

[12] Hendriks, H. (1987), "Type change in semantics: the scope of quantification and coordination", In: E. Klein & J. van Benthem (eds.), *Categories, Polymorphism and Unification*, Edinburgh/Amsterdam.

[13] Kamp, H. (1973), "Free choice permission", *Proceedings of the Aristotelian Society*, N.S., 74, pp. 57-74.

[14] Kamp, H. (1979), "Semantics versus pragmatics", In: F. Guenthner and J. Smidt (eds.), *Formal Semantics and Pragmatics for Natural Language*, Reidel, Dordrecht, pp. 225-78.

[15] Keenan, E. and L. Faltz (1985), *Boolean Semantics for Natural Language*, Kluwer, Dordrecht.

[16] Kratzer, A. (1978), *Semantik der Rede. Konteksttheorie - Modalwörter - Konditionalsätze*, Scriptor, Köningstein/Taunus.

[17] Kratzer, A. (1989), "An investigation of the lumps of thought", *Linguistics and Philosophy*, 12, pp. 607-653.

[18] Lewis, D. (1970/9), "A problem about permission", In: E. Saarinen et al. (eds.), *Essays in Honour of Jaakko Hintikka*, Reidel, Dordrecht (ms. 1970).

[19] Lewis, D. (1973), *Counterfactuals*, Oxford: Blackwell.

[20] Lewis, D. (1979), "Scorekeeping in a language game", *Journal of Philosophical Logic*, 8, pp. 339-359.

[21] Merin, A. (1992), "Permission sentences stand in the way of Boolean and other lattice- theoretic semantics", *Journal of Semantics*, 9, pp. 95-162.

[22] Meyer, J. (1993), "A different approach to deontic logic: Deontic Logic viewed as a variant of dynamic logic", *Notre Dame Journal of Formal Logic*, 29, pp. 109-136.

[23] Mott, P. (1973), "On Chisholm's paradox", *Journal of Philosophical Logic*, 2, pp. 197-211.

[24] Peleg, D. (1987), "Concurrent dynamic logic", *Journal of the Association for Computing Machinery*, 34, 2, pp. 450-479.

[25] Portner, P. (1997), "The semantics of mood, complementation, and conversational force", *Natural Language Semantics*, 5, pp. 167-212.

[26] Rohrbaugh, G. (1996), "An event-based semantics for deontic utterances", In: P. Dekker and M. Stokhof (eds.), *Proceedings of the 10th Amsterdam Coloquium*, Amsterdam.

[27] Rooy, R. van (1997), *Attitudes and Changing Contexts*, Ph.D. dissertation, University of Stuttgart.

[28] Rooy, R. van (1999), "Some analyses of pro-attitudes", In: H. de Swart (ed.), *Logic, Game theory and Social choice*, Tilburg University Press, Tilburg.

[29] Spohn, W. (1975), "An analysis of Hansson's dyadic deontic logic", *Journal of Philosophical Logic*, 4, pp. 237-252.

[30] Spohn, W. (1987), "Ordinal conditional functions: a dynamic theory of epistemic states", In: *Causation in Decision, Belief Change, and Statistics*, W.L. Harper and B. Skyrms (eds.), Dordrecht: Reidel, vol. 2, pp. 105-134.

[31] Stalnaker, R. (1975), "Indicative conditionals", *Philosophia*, 5, pp. 269-286.

[32] Stalnaker, R. (1978), "Assertion", In: P. Cole (ed.), *Syntax and Semantics, vol. 9: Pragmatics*, pp. 315-332.

[33] Stalnaker, R. (ms.), *Comments on Lewis's problem about permission.*

[34] Tan, Y. H. and L. van der Torre (1997), "Obligations in update semantics", In: K. van Marcke & W. Daelemans (eds.), *Proceedings of the Ninth Dutch Conference of Artificial Intelligence*, University of Antwerp, pp. 249-253.

[35] Veltman, F. (1996), "Defaults in update semantics", *Journal of Philosophical Logic*, 25, pp. 221-261.

[36] Winter, Y. (1995), "The square of individuals", In: P. Dekker and M. Stokhof (eds.), *Proceedings of the 10th Amsterdam Coloquium*, Amsterdam.

[37] Zimmermann, E. (1999), "Disjunction as conjunction" ..., In:... Salt.

20 Questions on Dialogue Act Taxonomies

David R. Traum
UMIACS
University of Maryland

Abstract

There is currently a broad interest in dialogue acts and dialogue act taxonomies, and new uses, taxonomies, and standardization efforts continue to be proposed. This paper presents a discussion of issues that are important to be addressed, if taxonomies are meant to be shared and understood the same way by proposers and others. The discussion is framed in terms of 20 questions, the answers to which will help make the meanings of taxonomy elements more clear to disjoint communities of users.

Introduction

There is currently a very wide range of theories and taxonomies of dialogue acts[1] available for a researcher to choose from. Moreover, specific deficits in any theory lead researchers to continue to develop new taxonomies to suit their particular purposes. To some degree, this is to be expected; dialogue act taxonomies can be seen as a kind of language for describing communicative events, and there is certainly no deficit in the continued creation and development of new formal languages (e.g., programming languages like Java) or (at a slower pace) natural languages. On the other hand, in both natural and artificial languages, the use of similar signs for different concepts can cause confusion and misunderstanding, often with serious undesirable consequences (e.g., in programming languages, the use of = as an assignment rather than equality operator in a boolean context; or the firing of an American city official for using the word *niggardly* (of independent Scandinavian origin) because it sounded too similar to an offensive racial epithet euphemistically referred to as "the N word"[2]. Similar confusions often occur when one researcher tries to interpret the dialogue act taxonomy of another. For example, when encountering an act labeled as *inform*: which subset of the constraints in (1) can said to be claimed (or perhaps none of them, depending on some other formulation entirely). This kind of confusion has led some (e.g., (External Interfaces Working Group, 1993; Discourse Resource Initiative, 1997; FIPA, 1997)) to propose standard theories that could be well-defined and understood and used across groups while others (e.g. (Allwood, 1977; Cohen and Levesque, 1990)) prefer to treat dialogue act (i.e., illocutionary force) identification as of only secondary importance, as a derived concept within a more general theory of rational interaction, using other primitives.

(1) a. declarative mood was used

b. propositional information was expressed

c. new information was expressed

d. the addressee came to believe what was expressed

e. what was expressed is actually believed by the speaker

f. what was expressed is actually true

It is hard to dispute that dialogue acts are a useful concept, given the wide variety of uses to which they are put. Some of these uses include[3]: representations of the pragmatic meaning of utterances in dialogue theories (Vanderveken, 1991; Bunt, 1996; Poesio and Traum, 1997; Poesio and Traum, 1998), building blocks for grammars of dialogue (Winograd and Flores, 1986; Bilange, 1991), labels for corpus annotation (Carletta et al., 1997; Alexandersson et al., 1997), agent communication languages (External Interfaces Working Group, 1993; Sidner, 1994; FIPA, 1997; Singh, 1998), object of analysis in dialogue systems (Allen et al., 1996; Bretier and Sadek, 1996), and element of a logical theory of rational interaction (Sadek, 1991). Despite this popularity of the concept, there are still a number of issues that present significant challenges for creating a taxonomy of dialogue acts that can be understood and used by other groups. Here, I will briefly raise some of the issues that have often caused confusion when interpreting one taxonomy of dialogue acts within the viewpoint of another. These issues must be addressed in order to have a clearer idea of what one means by saying a dialogue act occurred, whether the dialogue act taxonomy is meant for labeling a naturally occurring corpus, as part of a formal theory of action, or as a system-internal representation of the dialogue. Although there are many such issues and variations on the ones listed below, I focus here on 20, in the form of questions, in homage to the "dialogue game" named in the title. These questions are grouped, for convenience into sections of related questions.

Defining Dialogue Acts

1: Which is most important: fit to intuitions or formal rigor?

This question has implications beyond just dialogue act definitions, but is applicable for any attempt to provide a formal theory of commonsense notions. Very often it is difficult to precisely formulate complex intuitions using available formal techniques. The question then arises as to which goal to sacrifice for the time being: should one formalize a simpler notion that does not have the same properties of the intuitive concept (e.g., for belief, an undesirable property of logical omniscience, in a normal modal logic), or sacrifice some desirable formal properties, such as a model-theoretic semantics, necessary and sufficient conditions for categories, or soundness and completeness of an inference system. The answer will depend on the purposes to which the concept is to be put: if the primary goal is to discover and prove properties of the system, formal properties are not easily sacrificed. On the other hand, if the goals are more empirically motivated, a formal system with undesirable properties may not be close enough to be useful, while a concept without some of these properties may suffice for the task at hand (corpus labelling or use in a computer program). There should also be a place for intermediate points, that make some sacrifices at each side, while striving for maximum utility in a given purpose.

In particular, with respect to dialogue acts, it can be relatively easy to state precise definitional conditions of occurrence within a formal logic of action, however a problem may arise when these conditions diverge from a more intuitive (and intuitively useful) notion of action that empirical analysts and dialogue system designers would actually like to use.

2: Is the definition of a dialogue act an issue of Lexical Semantics or Ontology of Action?

There are different tasks one might be attempting when defining the meaning of a dialogue act. Is it to provide an account of when someone might be justified in describing an occurrence using a sentence headed with a particular verb (e.g., *inform*, *request*), or to provide a technical vocabulary to compactly describe various types of occurrences in convenient ways for analyzing other aspects of interaction. As (Allwood, 1977) warns, these endeavors should be clearly separated, even if one might want to use similar categories to describe each (as is done in (Allwood, 1980)), or maintain a position of identity of semantic and conceptual structure (Jackendoff, 1983). Intuitions, or annotation by naive coders without instructions to the contrary may tend to focus on the former enterprise, which may have undesirable consequences for the way in which the taxonomy is to be used. The key question is how much weight, if any, should be given to linguistic intuitions about when it is true or appropriate to use a particular sentence to describe an occurrence. For lexical semantics, this is the paramount question (barring issues of polysemy), while it might not really be a factor for an ontology which might diverge from language classifications for independently motivated reasons.

3: Under what conditions may an action be said to have occurred?

There are a number of different criteria that are being used to decide whether or not an action occurs in a given situation. (Allwood, 1980) uses four criteria, shown in (2), each of which can be a sufficient condition for ascribing that an action has occurred, while none is necessarily present.

(2) a. intention of performer

b. form of the behavior (e.g., linguistic form)

c. achieved result

d. context in which the behavior occurs

While it is certainly coherent to define actions in terms of meeting minimal conditions along any of these dimensions, it is less clear that this is the most useful way of capturing the generalities over acts that consumers of a dialogue act taxonomy would like to express. E.g., one may be interested strictly in the result, intention, or context, or perhaps between the relationship between form and result. In the most central case, all four kinds of conditions will hold, however one must know what to do when only some but not others hold. One should especially take care to avoid defining dialogue acts according to, say, a certain set of results holding, and then identify instances of these acts occurring strictly by one of the other criteria, leading to an incorrect claim of the results holding. Using different criteria (e.g., just results vs. intention, or vs. any of the four) can also lead to misunderstandings between theorists (or coders) as to whether a particular act has been performed, and whether the performance of an act implies a particular result holding.

As an example, consider a characterization of an *inform* act, given in (3).

(3) a. intention of performer: that receiver believes proposition **p**.

b. form of the behavior: speaker utters a declarative sentence with propositional content **p**.

c. achieved result: speaker and hearer mutually believe **p**.

d. context in which the behavior occurs: Speaker and hearer in contact, speaker believes **p**, hearer does not believe **p**. .

One could, of course, quibble with any of these characterizations in terms of being too strong or two weak to capture the meaning of "inform", or perhaps decide that they are more appropriate for some other act (e.g., *statement, assertion*). For example, one might produce an utterance of the same form, when not all of the conditions hold, or in which the speaker has a different intention.

Which kinds of conditions and whether only some or all of them might be necessary will also depend on the task being attempted. E.g., lexical semantics or action ontology, as in the previous question. Also, whether this ascription is made from the point of view of an online dialogue participant (such as a dialogue system) or an external observer, e.g., in labelling a corpus (see also question 6).

4: What is the role of speaker intention?

Intention is usually given a somewhat privileged position with respect to constituting dialogue acts (or action in general), e.g., the first criterion in (2). Some would define dialogue acts on the basis of the intention behind them, while others would equate illocutionary acts with recognition of this intention (based on the notion of meaning in (Grice, 1957)). A problem is that definitively interpreting the intention of the speaker requires mind-reading on the part of the hearer. Another problem is that some dialogue acts (like other acts) can be performed unintentionally or with an only ex post facto commitment. Finally, as with other acts, one may perform them with various goals in mind – it may be unnecessary to discover the actual intention in order to recognize an actor its effects in context. For example, a declarative utterance might be performed with the intention to cause the hearer to adopt a belief in the stated proposition, as in (3a). However, the same utterance might very well be performed if the speaker intends instead to cause the hearer to believe that the speaker believes p. Or that the speaker wants the hearer to believe p. Or the conjunction of some set of these (or other similar conditions).

For these reasons, some prefer to keep distinct (though related) the issues of intention recognition and dialogue act attribution.

5: What is the role of addressee uptake?

Regardless of speaker intention, many dialogue acts require for even the most limited notion of success some changes to the addressee based on understanding of the utterance in a particular way. Noticing whether the hearer has actually understood in a particular way can often require just as much mind-reading on the part of the speaker as intention recognition requires on the part of the hearer. Later utterances in a dialogue often provide more clues, and thus some, e.g., (Clark and Schaefer, 1989; Traum and Hinkelman, 1992) require a *grounding* process (in the later case by performing other kinds of dialogue acts) before considering some dialogue acts, such as *inform, request* to have been successfully performed. This involves the giving of positive and negative feedback (Allwood et al., 1992) about how utterances were perceived and understood.

A negotiation of meaning can also occur (McRoy and Hirst, 1995), severing completely the link between the dialogue effects and original speaker intentions or addressee uptake.

6: What point of view should be taken regarding performance of acts?

There are several points of view which may be taken when regarding the performance of dialogue acts. Relating to the previous two questions are the speaker's and hearer's point of view, respectively. Also, there is a *negotiated* collaborative point of view of the speaker-hearer team, which might differ from the private views of each of the participants. There is also a normative-conventional point of view, which can make reference to social institutions beyond just the speaker hearer pair. There is also the issue of time with respect to coding or ascription of acts: is it an on-line decision just at the time of performance, or is one allowed to view subsequent utterances/action, as well, before deciding what happened?

Point of view is relatively straightforward from a dialogue system internal perspective (though a system might still need to reason about the interlocutor's point of view And subsequent time points in diagnosing misunderstanding (McRoy and Hirst, 1995) or constructing a negotiated view), however it is far from clear what point of view should be taken by coders (and how they should estimate speaker or hearer point of view without mind-reading). Likewise, in defining the acts or giving them a logical semantics, point of view may be necessary to take into account.

As an example, consider the case of a feedback reply of a word or phrase following a declarative utterance by the other speaker. There are several different grounding functions that could be performed by this second utterance, such as *acknowledgement, repair, request for repair,* or *request for confirmation.* The latter two could perhaps be distinguished from the former by prosody: questioning intonation indicating lack of certainty and desire for further feedback, while declarative intonation indicating one of the former functions. One could distinguish acknowledgement from repair by deciding whether the second utterance repeats (or paraphrases) the former (acknowledgement) or replaces some part of it (repair). However, this decision requires a point of view for who believes it to be the same. Especially for current technology speech recognition systems, there is a significant likelihood that a system may repeat what it understood, which differs from what was actually said. It is also possible (though perhaps less likely) that a system intends to correct but ends up repeating what was really said. The same issues come up (though with less frequency) in human-human conversation.

Dialogue Act components

7: How are actions used in a logic?

In formal theories, actions are usually seen as transitions from states to states (or worlds to worlds), while dialogue acts are seen as special cases of actions (though see question 11). AI theories of action generally associate several sets with actions: a set of effects (constraints on the resulting state), a set of pre-conditions (constraints on the initial state), and decompositions (subactions that, performed together constitute the action).[4]

In terms of the categories given in (2), the effects corresponds to the achieved result, aspects of context and intention may be related to the pre-conditions, and the form of the behavior amounts to the decompositions. The AI theories of action generally include requirements on each of these aspects, so that the axioms in (4) hold (where X is an action type, Pre and $Effects$ are the preconditions and effects of this action type, and $prev$ now and $next$ are "consecutive" time points).[5] (4a) involves reasoning from felicitous performance to effects, (4b) involves reasoning from performance to preconditions having held, and (4c) involves reasoning from performance of subactions to main action. In addition, something like the schema in (5) is used (though in only an abductive or circumscriptive sense, rather than a sound axiom), for reasoning from subaction to intention ascription (plan recognition). Use of these kinds of axioms can also help determine inconsistency of a (default) interpretation, which may then be a cue of an indirect speech act, or a misunderstanding.

(4) a. $Pre(X, now) \wedge Try(X, now) \rightarrow Effects(X, next)$

b. $Done(X, now) \rightarrow Pre(X, prev)$

c. $Pre(Y, prev) \wedge decomp(Y, \{X_1, \ldots, X_n,\}) \wedge \forall X_i : Done(X_i, now) \rightarrow Done(Y, now)$

(5) $Do(A, X) \wedge decomp(Y, \{\ldots, X, \ldots\}) \rightarrow Intend(A, Y)$

8: What kinds of information are the conditions and effects about?

Given the general framework for actions in the discussion of the previous question, there's still a large question of what aspects of the situation are relevant to defining conditions for dialogue act performance, and what kinds of things are (directly) affected. Some logical models might allow the truth value of any representable proposition to be a possible condition or effect. This must, of course, be filtered through the lens of "point of view" (see question 6). Generally there are three more special sorts of information used for conditions and effects of dialogue acts.

First, there is a notion of dialogue state, as encoded as state in a dialogue grammar (Winograd and Flores, 1986; Traum and Allen, 1992), or other structural representation of context (Ginzburg, 1998). For example, certain acts may have as their pre-condition that the dialogue be in a particular state in the transition network, or, e.g., that an *answer* is possible only if there is a question under discussion. The effects can be a transition to a new state in the network, or other effects to the dialogue structure (see also question 12).

The second kind of information, the most popular in the planning approach, is in terms of mental states (e.g., belief, intention) of the speaker and addressee(s) (Cohen and Perrault, 1979; Allen and Perrault, 1980). For instance, pre-conditions of an *inform* act may include the latter two conditions in (3d). Effects will include newly adopted beliefs and intentions.

A third alternative is in terms of the social obligations and commitments undertaken by the dialogue participants (Poesio and Traum, 1998; Traum, 1999; Singh, 1998). Example effects include commitments to stated propositions, and commitments to do promised actions. Pre-conditions of this sort are more rare, though could include things like *excuses*, which presuppose a sort of obligation to act (which has not been or will not be performed).

Most approaches will actually combine two or three of these kinds of conditions and effects. There may also be other types of effects, not easily classifiable into these categories.

9: What kind of conditions are most appropriate ?

The notion of *pre-condition* is often criticized as meaning too many different things in relation to planning and reasoning about action (e.g., (Pollack, 1990). First of all, there is the general issue of enabling conditions vs. applicability constraints – the former being those that can be planned to achieve, while the latter describe conditions in which this kind of action should be considered. There is also the issue of whether these conditions are necessary or sufficient for (successful) performance of the action.

Many convenient dialogue acts actually have few if any actual pre-conditions, in the sense that the action can not occur if the conditions are not met. Conditions are often formed in terms of either normal conditions or in terms of what is required for felicitous performance of the action (Searle, 1969). Formulating conditions in this way does give greater flexibility, however at the expense of having to determine whether the conditions are felicitous (in addition to determining whether the action has been performed) and having to also describe *non-felicitous* performance.

The kinds of conditions to represent in a theory will also depend on the type of cognitive tasks to be performed using the acts: dialogue act planning and performance or recognition. For the former (e.g., using axiom (4a)), one might care more about sufficient rather than necessary conditions. However, for the latter (using e.g., axiom (4b)), one might be more interested in necessary conditions (to use this as an axiom rather than default rule).

10: How should an unsuccessful act be distinguished from a failed attempt to perform an act?

This question is related to the difference between *success* and *satisfaction* of a speech act (Vanderveken, 1990). The former has to do with whether the act was actually fully performed, the latter with whether the propositional content is (or becomes) true. If one uses a social commitment approach, then one may say the act has been performed if the commitments are established, and (fully) successful if its intended perlocutionary effects (Sadek, 1991) or evocative intentions (Allwood, 1995) are achieved.

As an example, consider a request by A to B, for B to do some action `x`, schematically: `Request(A,B,Do(B,x))`. The question becomes which of the conditions in (6) do we associate with an attempt vs success vs. satisfaction? Condition (6f) seems sufficient for an attempt, while (6a) is necessary for full satisfaction. Success criteria are more difficult to agree upon, however (see also question (8)). According to the mental states approach, successful performance of the request might be (6b), or (6e), with an additional assumption of cooperativity to lead to (6b) and then (6a) (Cohen and Perrault, 1979). The social commitments approach would favor (6c) (Allwood, 1994), or (6d) (Traum, 1994), with (6c) coming only on acceptance of the request.

```
(6)a. Do(B,x)
   b. Intend(B, Do(B,x))
   c. Obliged(B, Do(B,x))
   d. Obliged(B, Address(B, Act1))
   e. Believe(B, Want(A,Do(B,x)))
   f. Try(A,Request(A,B,Do(B,x)))
```

Another issue is what kinds of actions are involved in leading to success of the action (and the associated effects)? Is a single utterance (in the appropriate circumstances) enough, or is a grounding process (Clark and Schaefer, 1989; Traum, 1994) needed

Relationships and Complex Acts

11: What is the relationship between dialogue acts and other (e.g., physical) acts?

One of the main intuitions behind speech act theory (doing things with words (Austin, 1962)) was to connect speech acts with other actions. However different theories may maintain a crisp or more blurred distinction between dialogue acts and non-communicative acts. Some want a clear distinction, while others would want to use the same logic of action account to account for both. (Litman and Allen, 1987) distinguished dialogue acts as being *meta-acts*, defining discourse plans as having other plans (domain or discourse) as parameters. (Lambert and Carberry, 1991) also distinguish discourse, domain and problem solving plans and actions.

Depending on the answer to question 8, some may want to describe dialogue acts as having a different sort of effect on the dialogue context, mental states, or social context than can be achieved with other kinds of action.

12: What is the relationship between dialogue acts and dialogue structure?

There are several options. Some conceive dialogue structure as being wholly dependent on the structure of dialogue acts performed (e.g., grammar-based approaches like (Sinclair and Coulthard, 1975)) Others use a different sort of structure, not directly composed of dialogue act performance to represent things like accessibility, topic and focus, and global coherence, which can be sensitive to other aspects of the utterances, or be primarily constructed from the activity that the participants are engaged in (Allwood, 1995; Grosz and Sidner, 1986). In this latter case, it remains to be explicated what effect (if any) performance of different kinds of dialogue acts have on this dialogue structure. Dialogue structure is also often used as one of the aspects of context for dialogue act performance, which can serve as the source of pre-conditions and input for action recognition.

13: Are there multi-agent dialogue acts?

As mentioned related to question 5, some see the performance of most illocutionary acts as the collective performance of multiple agents, in virtue of the grounding process. Other candidates for multi-agent action include notions of higher-level activity such as *games* (Severinson Eklundh, 1983) or *exchanges* (Sinclair and Coulthard, 1975), or collaborative completions where one speaker finishes another's sentence. There are several difficulties with these kinds of acts, however. First is related to reliable tagging and computing proper inclusion/exclusion of relevant parts of the collaborative action. Finding the right "units" at which to apply the tags can be a difficult process (see, e.g., discussions in (Discourse Resource Initiative, 1997; Nakatani and Traum, 1999)). This difficulty is compounded when there are multiple acts with different boundaries (e.g., the single-agent act and multi-agent component performed by a speaker within an utterance).

Another issue is the kind of logic that will allow this kind of action will need to be more complex than that required for single agent action.

14: Can dialogue acts be "composed" of more primitive acts

If a dialogue act taxonomy has multiple strata of acts, then the question becomes whether these strata are conceived of as *levels* or *ranks*, according to the terminology of (Halliday, 1961), that is, whether there could be some grammar or recipe for performance of an act of one stratum using acts of a lower stratum, in the way that sentences can be composed of words and phrases (rank), or whether these are different kinds of phenomena, like the distinction between phonology and syntax (level). For example, the 4 tiered system in (Sinclair and Coulthard, 1975) is conceived of as ranks within a general "discourse" level, and e.g., the *check game* in the Maptask coding scheme (Carletta, 1992) is composed of an initiating check moves, along with other moves that accomplish the purpose of the check. On the other hand, the multi-tiered system in (Traum and Hinkelman, 1992) is conceived of in terms of ranks (at least for the lower three levels), and, although core speech acts like *inform* are only successfully realized at the point of a completed structure of *grounding acts*, there is no relationship between the type or sequence of grounding acts performed and the type of core speech acts which are realized.

Within the plan ontology described related to question 7, this amounts to a question of whether the decomposition of a dialogue act can contain other dialogue acts, or some other sort of realization.

15: Can multiple dialogue acts occur at the same time (performed through the same utterance)?

Since most utterances have multiple functions, the answer, given most definitions according to conditions and effects, will be "yes." However there are a number of complications, depending on the use to which the taxonomy is put. For logical theories, one important question is whether the logic can accommodate simultaneous action or *level-generation* (Goldman, 1970). Simple versions of, e.g. the situation calculus (McCarthy and Hayes, 1969) or dynamic logic (Harel, 1979) do not, which makes it difficult to formalize this kind of phenomenon. Likewise, within dialogue systems, reasoning about act occurrence is often made not on the basis of necessary and sufficient conditions, but on closeness of fit, using abductive or statistical methods. Such methods generally are used to decide on a particular label to the exclusion of others, e.g., that an interrogative utterance is an indirect request but not a question. Finally, in tagging a corpus, it is often tedious and unreliable to try to code all possible occurrences of a particular function, and so instructions are designed so as to only code the most significant (in the opinion of the coding task designer), e.g., the *code high* principle in (Condon and Cech, 1992). It is important to be explicit about such assumptions, and whether multiple dialogue acts are assumed to be allowed to happen at the same time, and what the meaning of something not being coded is: non-occurrence or no statement about occurrence or non-occurrence. In the Condon-Cech scheme, one could deduce that a "higher" act had not occurred, but no such deduction is warranted about a "lower" act.

Taxonomic Considerations

16: Can the same taxonomy be used for different kinds of activities?

There are two relevant notions of activity here. First is the meta-activity of recognizing or coding dialogue acts, that is the concern of question 20. Relevant types of activities include logical deduction, system participation, and corpus analysis. There is also the issue of on-line or off-line coding and amount of lookahead (see question 6). Here I will concentrate on the activities that the dialogue participants are engaged in.

There are a number of different dialogue activities that people are interested in designing taxonomies of dialogue acts for. Some examples include casual conversation (Jurafsky et al., 1997), classroom discourse (Sinclair and Coulthard, 1975), and various flavors of task-oriented dialogue, such as information seeking (van Vark et al., 1996), collaborative scheduling (Alexandersson et al., 1997), and direction following (Carletta et al., 1997)).

Taxonomies designed for different tasks or genres of dialogue tend to be quite different (e.g., even within the general realm of task-oriented cooperative dialogue, meeting-scheduling vs. direction following). To some extent, this is to be expected, since different genre's will have different frequencies of acts. For example, roughly 50% of utterances are *statements* in the Switchboard corpus, which is concerned with causal conversation (Jurafsky et al., 1998), while the HCRC Maptask corpus , concerned with instruction giving/following (Carletta et al., 1997), has only 8% of utterances labelled with the equivalent tag, *explain*.[6] Conversely, Maptask has 15.6% of utterances marked as *instruct*, while Switchboard has less than 1% of utterances labelled as *action-directive*. Interestingly, though, both have around 20% of utterances marked as *acknowledgement*. Different tasks and coding purposes may also place different demands on specificity of a taxonomy (see question 18), e.g., to have an appropriate reliability and perplexity for a given coding purpose.

Some hope that these different task specific "sub-taxonomies" might be fit together within a coherent general taxonomy of acts in dialogue. A general theory might also better allow one to identify activities as well as episodes within an activity and genres of activities. The DRI group has been working toward schemes that might have more general applicability (at least within the general category of task-oriented dialogue) (Discourse Resource Initiative, 1997; Core et al., 1999)). The SLSA project at Gothenburg University is investigating more generally the issue of corpus collection and dialogue coding of spoken language activities (Allwood, 1999).

17: Can the same taxonomy be used for different kinds of agents?

Related to the above question, is one of whether the same taxonomy could cover situations of humans communicating with humans, humans with machines, and machines with machines. Other possibilities could also include humans with animals or animals with animals (or animals with machines?). Even when only humans are communicating, there is still an important issue of the medium, e.g., face to face, spoken language only, multi-modal computer mediated communication of various flavors. These issues will certainly have a bearing on the distribution of act types. E.g., much more explicit grounding in spoken dialogue ($>$ 95% (Traum and Heeman, 1997)) than computer chat ($\sim$ 40% (Dillenbourg et al., 1997)) [7], and more explicit verifications from computer systems with relatively poor speech recognition than between fluent humans.

Again, the hope of many researchers is that the same taxonomies (at a suitably abstract level, concerning some of the lack of subtlety of machine communication) could be used concerning any of these sets of agents. Some, however, have pointed to the differences in communication styles between human-human and human-machine communication as a reason for anticipating different taxonomies, and not carrying over the insights from one to the other (Jönsson, 1995).

18: How detailed should a dialogue act taxonomy be?

There are many subtle gradations in speech act verbs, often relating to different facets of the participants or normative attitudes towards the content of the act (e.g., state, assert, inform, confess, concede, maintain, ...). The question arises as to how many of these distinctions should be captured within a dialogue act taxonomy. One key issue is whether one wants to capture generalizations or distinctions. There is also often a trade-off between precisely capturing differences in conditions and effects and confidence in a label.

If possible it may be best to arrange these fine distinctions within a hierarchical or lattice structure (as is done by, e.g., (Allen and Core, 1997; Alexandersson et al., 1997)), so that a degree of specificity may be chosen appropriate to the particular task. One issue is whether theorists and coders can agree on the hierarchical structure of related acts, which, in some cases, may be more controversial than the base labels themselves.

19: Where should complexity be realized in a coding taxonomy?

Given that utterances in dialogue are generally multi-functional, the question arises as to how best to capture this multiplicity of functions in a taxonomy. There are two extremes: one is to separate out each function and code it separately, requiring multiple labels for each utterance, one for each function. The advantage is fairly simple act definitions, each with fairly clear semantics and ascription conditions. The disadvantage is a large number of tagging decisions — one for each functional dimension, leading to a fairly onerous tagging task, and lower reliability on some dimensions depending on coder attention and atunement to each phenomenon. This approach is taken by (Discourse Resource Initiative, 1997).

The other extreme is to combine sets of coherent bundles into complex labels and code with these. The advantage is a potentially easier and more reliable coding task, especially if the same bundles appear repeatedly within a given coding effort. The disadvantage is that there might be many possible acts, if many gradations appear. If only some of these are assigned labels, then it may be difficult to decide how to code an utterance that shares some (but not all) of the features of one label, while having some features from another. This approach can also lead to missing connections between different acts that share some of the features, making it hard to analyze existence of these features from the coded data. This approach may be typified by the first Verbmobil coding scheme (Jekat et al., 1995).

It is possible both to find taxonomies that take a more middle position than either extreme, and that capture some of the advantages of the other scheme (while lessening the disadvantages of their own). For example, the Switchboard DAMSL scheme uses many ideas from (Discourse Resource Initiative, 1997; Allen and Core, 1997), while moving toward the other extreme of coding in discrete, mutually exclusive bundles rather than multiples dimensions. There are also proposals to do this for the main DRI scheme as well (Core et al., 1999). These schemes will still retain the theoretical connection to the multi-layer scheme, and so will be more easy to determine individual functions. Likewise, it should be possible to define optional rather than mandatory *macros* which combine convenient bundles of features into simplified coding tasks, while still maintaining the full flexibility of the multiple layer approach. This is the method advocated in (Cooper et al., 1999).

20: Can a taxonomy used for tagging dialogue corpora be given a formal semantics and/or used in a dialogue system?

The hope of many researchers is definitely a "yes" answer to this question: the purpose of tagging or formal semantics is often for use within a dialogue system. Moreover a clear semantics may help one to formulate sharper principles for a tagging exercise. There are some difficulties, however. One is the issue of different resources - one may require details of the content of an act in order to use in a system or provide semantics, yet this may be too onerous for a tagging exercise. Likewise, formal representations of context built from incorporation of previous acts may not be available during a coding task. On the other hand, human coders may be able to use complex intuitions in their coding which are difficult to incorporate in a formal description or implementation (though these may perhaps be learned from a corpus, using machine learning techniques (Reithinger and Klesen, 1997; Samuel, 1998)). These different skill sets may tend to make taxonomies designed for different purposes diverge.

Discussion

Given that the above questions are not exhaustive or binary, and have remained mostly at the meta-level, we can certainly see that formulating the ultimate dialogue act taxonomy is a much harder problem than the game of 20-questions. The discussion above is also far from the last word on any of these topics. The hope is that further research may yield some more definitive answers or at least better understanding of the issues involved. Meanwhile, the above discussion may help dialogue act theorists be clearer about some of the meanings of their taxonomy, in the hopes of wider understanding and applicability of the taxonomies that are used.

Acknowledgments

The author was supported during the writing of this paper by the TRINDI (Task Oriented Instructional Dialogue) project, EU TELEMATICS APPLICATIONS Programme, Language Engineering Project LE4-8314. I would also like to (anonymously) thank the many colleagues who helped me formulate my ideas on the topics discussed in this paper, and Jens Allwood and William Mann for helpful discussions on previous versions of this paper.

DAVID R TRAUM
UMIACS
A. V. Williams Building
University of Maryland
College Park, MD 20742 USA

Notes

[1] By the term *dialogue acts*, I don't mean to limit discussion to those theories and taxonomies that explicitly use this term. Other terms used for the same general concept include *locutionary, illocutionary, and perlocutionary acts* (Austin, 1962), *speech acts* (Searle, 1969), *communicative acts* (Allwood, 1976; Sadek, 1991; Airenti et al., 1993),

conversation acts (Traum and Hinkelman, 1992) and *conversational moves* (Carletta et al., 1997). My remarks here are intended to apply to the general phenomenon described by this range of terms. *Dialogue acts* can perhaps be seen as most generic, at least in the context of a workshop on dialogue.

[2]Washington D.C. Public Advocate David Howard, in February 1999.

[3]Here and elsewhere in the papers, examples are meant to be representative rather than exhaustive; there is a large amount of work in some of these areas.

[4](Pollack, 1990) focuses instead on *enabling conditions* rather than pre-conditions, and *generation conditions* rather than decomposition, (following (Goldman, 1970).

[5]Details of axioms of this sort obviously vary quite a bit depending on the syntax and semantics of the logic used, e.g., whether *Done* means "happened in the immediately prior state transition" or some looser sense of happened recently.

[6]Maptask Statistics from personal communication from Amy Isard.

[7]Although these studies concerned different tasks.

References

Airenti, G., Bara, B. G., and Colombetti, M. (1993). Conversation and behavior games in the pragmatics of dialogue. *Cognitive Science*, 17:197–256.

Alexandersson, J., Buschbeck-Wolf, B., Fujinami, T., Maier, E., Reithinger, N., Schmitz, B., and Siegel, M. (1997). Dialogue acts in VERBMOBIL-2. Verbmobil Report 204, DFKI, University of Saarbruecken.

Allen, J. and Core, M. (Draft, 1997). Draft of damsl: Dialog act markup in several layers. available through the WWW at: http://www.cs.rochester.edu/research/trains/annotation.

Allen, J. F., Miller, B. W., Ringger, E. K., and Sikorski, T. (1996). A robust system for natural spoken dialogue. In *Proceedings of the 1996 Annual Meeting of the Association for Computational Linguistics (ACL-96)*, pages 62–70.

Allen, J. F. and Perrault, C. R. (1980). Analyzing intention in utterances. *Artificial Intelligence*, 15(3):143–178.

Allwood, J. (1976). *Linguistic Communication as Action and Cooperation.* PhD thesis, Göteborg University, Department of Linguistics.

Allwood, J. (1977). A critical look at speech act theory. In Dahl, Ö., editor, *Logic, Pragmatics and Grammar.* Studentlitteratur.

Allwood, J. (1980). On the analysis of communicative action. In Brenner, M., editor, *The Structure of Action.* Basil Blackwell. Also appears as Gothenburg Papers in Theoretical Linguistics 38, Dept of Linguistics, Göteborg University.

Allwood, J. (1994). Obligations and options in dialogue. *Think Quarterly*, 3:9–18.

Allwood, J. (1995). An activity based approach to pragmatics. Technical Report (GPTL) 75, Gothenburg Papers in Theoretical Linguistics, University of Göteborg.

Allwood, J. (1999). The Swedish Spoken Language Corpus at Göt eborg University. In Andersson, R., Abelin, Å., and an d Per Lindblad, J. A., editors, *Fonetik99: Proceedings from the Twelfth Swedish Phonetics Conference*, number 81 in Gothenburg Papers in Theoretical Linguistics, pages 5–9. Department of Linguistics, Göteborg University.

Allwood, J., Nivre, J., and Ahlsen, E. (1992). On the semantics and pragmatics of linguistic feedback. *Journal of Semantics*, 9.

Austin, J. A. (1962). *How to Do Things with Words.* Harvard University Press.

Bilange, E. (1991). A task independent oral dialogue model. In *Proceedings of the Fifth Conference of the European Chapter of the Association for Computational Linguistics*, pages 83–88.

Bretier, P. and Sadek, M. D. (1996). A rational agent as the kernel of a cooperative spoken dialogue system: Implementing a logical theory of interaction. In Müller, J. P., Wooldridge, M. J., and Jennings, N. R., editors, *Intelligent Agents III — Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages (ATAL-96)*, Lecture Notes in Artificial Intelligence. Springer-Verlag, Heidelberg.

Bunt, H. (1996). Interaction management functions and context representation requirements. In *Proceedings of the Twente Workshop on Language Technology: Dialogue Management in Natural Language Systems (TWLT 11)*, pages 187–198.

Carletta, J. (1992). *Risk-taking and Recovery in Task-Oriented Dialogue.* PhD thesis, University of Edinburgh.

Carletta, J., Isard, A., Isard, S., Kowtko, J. C., Doherty-Sneddon, G., and Anderson, A. H. (1997). The reliability of a dialogue structure coding scheme. *Computational Linguistics*, 23(1):13–31.

Clark, H. H. (1992). *Arenas of Language Use.* University of Chicago Press.

Clark, H. H. and Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science*, 13:259–294. Also appears as Chapter 5 in (Clark, 1992).

Cohen, P. R. and Levesque, H. J. (1990). Rational interaction as the basis for communication. In Cohen, P. R., Morgan, J., and Pollack, M. E., editors, *Intentions in Communication.* MIT Press.

Cohen, P. R. and Perrault, C. R. (1979). Elements of a plan-based theory of speech acts. *Cognitive Science*, 3(3):177–212.

Condon, S. and Cech, C. (1992). Manual for coding decision-making interactions. unpublished manuscript, updated May 1995, available at: ftp://sls-ftp.lcs.mit.edu/pub/multiparty/coding_schemes/condon.

Cooper, R., Larsson, S., Matheson, C., Poesio, M., and Traum, D. (1999). Coding instructional dialogue for information states. Deliverable D1.1, Trindi Project.

Core, M., Ishizaki, M., Moore, J., Nakatani, C., Reithinger, N., Traum, D., and Tutiya, S. (1999). The report of the third workshop of the discourse resource initiative, chiba univeristy and kazusa academia hall. Technical Report No.3 CC-TR-99-1, Chiba Corpus Project.

Dillenbourg, P., Jermann, P., Schneider, D., Traum, D., and Buiu, C. (1997). The design of moo agents: Implications from a study on multi-modal collaborative problem solving. In *Proceedings of the 8th World Conference on Artificial Intelligence in Education (AI-ED 97)*, pages 15–22.

Discourse Resource Initiative (1997). Standards for dialogue coding in natural language processing. Report no. 167, Dagstuhl-Seminar.

External Interfaces Working Group (1993). Draft specification of the kqml agent-communication language. available through the WWW at: http://www.cs.umbc.edu/kqml/papers/.

FIPA (1997). Fipa 97 specification part 2: Agent communication language. working paper available at http://drogo.cselt.stet.it/fipa/spec/fipa97/f8a21.zip.

Ginzburg, J. (1998). Clarifying utterances. In Hulstijn, J. and Niholt, A., editors, *Proc. of the Twente Workshop on the Formal Semantics and Pragmatics of Dialogues*, pages 11–30, Enschede. Universiteit Twente, Faculteit Informatica.

Goldman, A. I. (1970). *A Theory of Human Action.* Prentice Hall Inc.

Grice, H. P. (1957). Meaning. *Philosophical Review*, 66:377–88.

Grosz, B. J. and Sidner, C. L. (1986). Attention, intention, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Halliday, M. A. K. (1961). Categories of the theory of grammar. *Word*, 17:241–92.

Harel, D. (1979). *First Order Dynamic Logic*. Springer-Verlag.

Jackendoff, R. (1983). *Semantics and Cognition*. The MIT Press.

Jekat, S., Klein, A., Maier, E., Maleck, I., Mast, M., and Quantz, J. (1995). Dialogue Acts in VERBMOBIL. Technical Report 65, BMBF Verbmobil Report.

Jönsson, A. (1995). Dialogue actions for natural language interfaces. In *Proc. of the 14th IJCAI*, pages 1405–1411, Montreal, Canada.

Jurafsky, D., Shriberg, E., and Biasca, D. (1997). Switchboard swbd-damsl shallow-discourse-function annotation coders manual. Technical Report 97-02, University of Colorado Institute of Cognitive Science. Draft 13.

Jurafsky, D., Shriberg, E., Fox, B., and Curl, T. (1998). Lexical, prosodic, and syntactic cues for dialog acts. In *Proceedings of ACL/COLING-98 Workshop on Discourse Relations and Discourse Markers*, pages 114–120, Montreal, Canada.

Lambert, L. and Carberry, S. (1991). A triparite plan-based model of discourse. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 47–544.

Litman, D. J. and Allen, J. F. (1987). A plan recognition model for subdialogues in conversation. *Cognitive Science*, 11:163–200.

McCarthy, J. and Hayes, P. (1969). Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B. and Michie, D., editors, *Machine Intelligence 4*, pages 463–502. Edinburgh University Press. Also appears in N. Nilsson and B. Webber (editors), *Readings in Artificial Intelligence*, Morgan-Kaufmann.

McRoy, S. W. and Hirst, G. (1995). The repair of speech act misunderstandings by abductive inference. *Computational Linguistics*, 21(4):5–478.

Nakatani, C. H. and Traum, D. R. (1999). Coding discourse structure in dialogue (version 1.0). Technical Report UMIACS-TR-99-03, University of Maryland.

Poesio, M. and Traum, D. R. (1997). Conversational actions and discourse situations. *Computational Intelligence*, 13(3).

Poesio, M. and Traum, D. R. (1998). Towards an axiomatization of dialogue acts. In *Proceedings of Twendial'98, 13th Twente Workshop on Language Technology: Formal Semantics and Pragmatics of Dialogue*.

Pollack, M. E. (1990). Plans as complex mental attitudes. In Cohen, P. R., Morgan, J., and Pollack, M. E., editors, *Intentions in Communication*. MIT Press.

Reithinger, N. and Klesen, M. (1997). Dialogue act classification using language models. In *Proc. Eurospeech '97*, pages 2235–2238, Rhodes, Greece.

Sadek, M. D. (1991). Dialogue acts are rational plans. In *Proceedings of the ESCA/ETR workshop on multi-modal dialogue*.

Samuel, K. (1998). DISCOURSE LEARNING: Dialogue act tagging with transformation-based learning. In *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98) and of the 10th Conference on Innovative Applications of Artificial Intelligence (IAAI-98)*, pages 1199–1199, Menlo Park. AAAI Press.

Searle, J. R. (1969). *Speech Acts*. Cambridge University Press, New York.

Severinson Eklundh, K. (1983). The notion of language game – a natural unit of dialogue and discourse. Technical Report SIC 5, University of Linköping, Studies in Communication.

Sidner, C. L. (1994). An artificial discourse language for collaborative negotiation. In *Proceedings of the forteenth National Conference of the American Association for Artificial Intelligence (AAAI-94)*, pages 814–819.

Sinclair, J. M. and Coulthard, R. M. (1975). *Towards an analysis of Discourse: The English used by teachers and pupils.* Oxford University Press.

Singh, M. P. (1998). Agent communication languages: Rethinking the principles. *IEEE Computer*, 31(12):40–47.

Traum, D. R. (1994). *A Computational Theory of Grounding in Natural Language Conversation.* PhD thesis, Department of Computer Science, University of Rochester. Also available as TR 545, Department of Computer Science, University of Rochester.

Traum, D. R. (1999). Speech acts for dialogue agents. In Rao, A. and Wooldridge, M., editors, *Foundations of Rational Agency*. Kluwer.

Traum, D. R. and Allen, J. F. (1992). A speech acts approach to grounding in conversation. In *Proceedings 2nd International Conference on Spoken Language Processing (ICSLP-92)*, pages 137–40.

Traum, D. R. and Heeman, P. (1997). Utterance units in spoken dialogue. In Maier, E., Mast, M., and Luperfoy, S., editors, *Dialogue Processing in Spoken Language Systems — ECAI-96 Workshop*, Lecture Notes in Artificial Intelligence, pages 125–140. Springer-Verlag, Heidelberg.

Traum, D. R. and Hinkelman, E. A. (1992). Conversation acts in task-oriented spoken dialogue. *Computational Intelligence*, 8(3):575–599. Special Issue on Non-literal language.

van Vark, R., de Vreught, J., and Rothkrantz, L. (1996). Analysing ovr dialogue coding scheme 1.0. Technical Report 96-137, TU Delft Faculty of Technical Mathematics and Informatics.

Vanderveken, D. (1990). On the unification of speech act theory and formal semantics. In Cohen, P. R., Morgan, J., and Pollack, M. E., editors, *Intentions in Communication.* MIT Press.

Vanderveken, D. (1990-1991). *Meaning and Speech Acts.* Cambridge University Press.

Winograd, T. and Flores, F. (1986). *Understanding Computers and Cognition.* Addison-Wesley Publishing Company, Inc.

Understanding Mathematical Discourse

Claus Zinn
Lehrstuhl für Künstliche Intelligenz (IMMD VIII)
Universität Erlangen-Nürnberg

Abstract

Discourse Understanding is hard. This seems to be especially true for mathematical discourse, that is proofs. Restricting discourse to mathematical discourse allow us, however, to study the subject matter in its purest form. This domain of discourse is rich and well-defined, highly structured, offers a well-defined set of discourse relations and forces/allows us to apply mathematical reasoning.

We give a brief discussion on selected linguistic phenomena of mathematical discourse, and an analysis from the mathematician's point of view. Requirements for a theory of discourse representation are given, followed by a discussion of proofs plans that provide necessary context and structure. A large part of semantics construction is defined in terms of proof plan recognition and instantiation by matching and attaching.

1 Introduction

Human-based proof verification. Understanding a mathematical discourse means to being able to verify the correctness of the given mathematical argument. Verifying a proof is by no means trivial and often time consuming. As DAVIS shows, only some mathematicians would volunteer to check a fifty-page proof [7]. Often, only a small number of mathematicians of some sub-discipline have the necessary qualification to check a proof of their domain completely for correctness. The verification process itself is error-prone. It is therefore inevitable that even many published proofs are either incomplete and/or error-prone.

Verifying a proof is also of social nature [9]:

> *"The acceptance of a theorem by practising mathematicians is a social process which is more a function of understanding and significance than of rigorous proof."*

HANNA gives some criteria why mathematicians accept a proof [9, p. 70]:

1. *They understand the theorem, the concepts embodied in it, its logical antecedents, and its implications. There is nothing to suggest it is not true;*
2. *The theorem is significant enough to have implications in one or more branches of mathematics (and is thus important and useful enough to warrant detailed study and analysis);*
3. *The theorem is consistent with the body of accepted mathematical results;*
4. *The author has an unimpeachable reputation as an expert in the subject matter of the theorem.*

5. *There is a convincing mathematical argument for it (rigorous or otherwise), of a type they have encountered before.*

If there is a rank order of criteria for admissibility, then these five criteria all rank higher than rigorous proof.

HANNA argues further that the *"mathematician is much more interested in the message embodied in the proof than its formal codification and syntax. The mechanics of proof are seen as a necessary but ultimately less significant aspects of mathematics. Certainly being able to follow the steps of a proof is not the same as understanding it"*. HANNA then cites Bourbaki[1]:

> *"A proof is not really 'understood' as long as he (the mathematician) has only verified the correctness of the deductions involved step by step, without trying to understand clearly the ideas which led to the construction of this chain of deductions in preference to all others"*

In 1976, ULAM estimated the number of published theorems and their proofs by 200.000 [19]. According to Hanna [9], only a small part of these theorems get recognized by the mathematical community and accepted as true statements. Will we have, in which distant future, computer programs which do the proof reading, or which assist mathematicians in doing so?

Machine-based proof verification. Imagine a machine that is capable to understand mathematical discourse. You can give a natural language proof for some theorem to the machine, it then analyses the input and communicates its knowledge about what is has read, analysed and understood. Of course, building a machine that understands discourse is one of the main goals of Natural Language Processing. Restricting discourse to mathematical discourse allows us, however, to study the subject matter in its purest form. Fig. 1 depicts the main characteristics of mathematical discourse.

+ well-defined and rich domain of discourse
+ highly structured mathematical discourse
+ well-defined and small set of discourse relations
+ mathematical reasoning

Figure 1: Mathematical discourse

Building an ontology for a mathematical domain, say elementary number theory, is much easier than for a more natural domain of discourse, say, for the reservation/booking domain. This is because mathematicians are used to define concepts before using them — respecting certain standards on *how to define*. A mathematical discourse is a highly structured form of discourse. Mathematicians agree on a variety of different methods for *how to prove* theorems. Often, the form and content of the theorem clearly indicate applicable proof methods. From the linguistic point of view, these proof methods define discourse plans which serve as good estimates on what to expect next, and therefore, to a large extent, facilitate the task of discourse understanding. Equally, discourse markers help to identify the logical relations that hold between parts of the proof. To being able to follow the reasoning line of the proof author, it is necessary to model mathematical reasoning itself — a task, however, which we consider much easier than modeling common sense reasoning of everyday discourses.

But what does it mean to understand a mathematical argument? In contrast to Hanna's criteria, we propose two answers:

Operational: the proof understander is able

- to answer questions about the proof accurately;
- to extract the main proof idea;
- to detect gaps and flaws in the argumentation line;
- to give a more rigorous proof.

Formal: the proof understander is able to generate a formal proof by using the informal mathematical argument.

Note that it might well be the case that the "formal semantics" is a prerequisite for implementing the "operational semantics".

Formal proofs vs. informal proofs. It is interesting to compare informal proofs taken from textbook proofs with formal proofs, generated automatically/interactively by/with a proof system. This gives us first ideas which difficulties one has to face.

(1) A formal proof is written in a formal language, the latter can be easily described in less than one page. An informal proof is written in an informal language, say English, having a large set of stylised syntactic constructions[2].

(2) In formal proofs there is no ambiguity. In informal proofs there should be no ambiguity, but there is.

(3) In a formal proof, for each proof step, it is explicitly given which conclusion is derivable by which set of premises and by which inference rule. In an informal proof, many proof steps are omitted or incomplete (incomplete set of premises, and lack of reference to inference rules used).

(4) Finding informal or formal proofs is not trivial.

(5) In a formal proof, the theory in which the proof is stated is explicitly stated. The theory, and nothing else, defines the context. In an informal proof, there is no full and explicit theory that one can refer to. The context is to be completed by the proof reader.

(6) Formal proofs are structured (e.g., resolution graphs, natural deduction trees, semantics tableaux). Informal proofs are structured, too (e.g., a proof per induction consists of induction base case, induction hypothesis and induction step, the first and the latter contain subproofs themselves).

(7) In a formal proof, form matters. In an informal proof, meaning matters.

(8) Machines are good to verify formal proofs, but currently cannot check informal proofs. Humans are good to verify informal proofs, and bad in verifying formal proofs.

In the remainder of this article, I will (i) discuss a selection of linguistic phenomena that occur in informal proofs; (ii) analyse informal proofs mathematically; (iii) define requirements for a theory of discourse representation capturing the specialties of mathematical discourse; (iv) introduce proof plans that introduce necessary context and structure for proof understanding.

2 An analysis of informal proofs

In this section, we give an analysis of mathematical discourse from both the linguistic and mathematical point of view.

2.1 A linguistic analysis of informal proofs

We discuss only problems that are typical for informal mathematical discourse.

2.1.1 Free and bound, anonymous and named variables

Variables and unknowns. In predicate logic, to determine if a variable is free or bound, we only have to analyse the form of the statement, and not at all its meaning. So, in $\forall y : f(x, y)$ the variable x is free, and the variable y is bound. In ordinary mathematics, from the statement alone, there is no way to tell if x is read to be a variable or an unknown. For example, in ordinary mathematics, the statement $x^2 - 1 = (x + 1)(x - 1)$ is true for all values of x, and normally, one would assume x being used as a variable. In the statement $x^2 - 4x + 3 = 0$ one would assume x being used as an unknown, and its value is to be determined. The matter gets complicated by a logical principle which allows unknowns become variables and vice versa.

Look at the following discourse.

(1)

> THEOREM 3 (EUCLID'S FIRST THEOREM). *If p is prime, and $p \mid ab$, then $p \mid a$ or $p \mid b$.*
> Suppose that p is prime and $p \mid ab$. If $p \not| a$ then $(a, p) = 1$, and therefore, by Theorem 24, there are an x and a y for which $xa + yp = 1$ or
>
> $$xab + ypb = b.$$
>
> But $p \mid ab$ and $p \mid pb$, and therefore $p \mid b$.

In this discourse, the p, a, and b that occur in the theorem are variables being universally quantified. But the p, a, b that occur later in the proof are unknowns. We could easily consistently rename all variables in the theorem without affecting the correctness of the proof (albeit the readability might be affected).

Note that the syntactic form of the statement that contains variables helps to determine if a variable occurs free, existentially bound, or universally bound. In 'there is an x and an y', both x and y are existentially bound. Sometimes, however, the form of the statement is misleading. For the theorem in discourse (1) gets a generic reading that is, all occurring variables get a universal binding. But for the third sentence of this discourse, 'If $p \not| a$ then $(a, p) = 1$', each of 'p', 'a', and 'b' occurs free.

Naming variables. Naming plays an important role in mathematical writing, and is partly determined by convention. In (1) we have several naming actions, for example, when asserting the existence of two natural numbers called x and y for which some equation holds. Expressing the same without the use of terms and formulae would be quite unreadable.

In [11], Lamport gives the following example:

(2) a. There do not exist four positive integers, the last being greater than two, such that the sum of the first two, each raised to the power of the fourth, equals the third raised to the same power.

b. There do not exist positive integers x, y, z, and n, with $n > 2$, such that $x^n + y^n = z^n$.

According to Lamport, (2a) reflects the style of seventeenth century mathematicians, and (2b) is the modern version. Variables are given names, and formulas are written in a more structured fashion. But names should only be introduced when necessary. As Krantz points out,

(3) a. Every nonnegative real number has a square root.

b. $\forall \mathrm{x} \exists \mathrm{y} : \mathrm{x} \geq 0 \Rightarrow \mathrm{y}^2 = \mathrm{x}$

(3a) is to be preferred to (3b). Mathematically, these two assertions are equivalent, but linguistically, they are quite different. Parsing the natural language form will, most probably result into a logical form that contains an unary predicate $square\_root$ and a binary predicate has, both of which cannot be found in (3b).

2.1.2 Terms, and means to reference them

As we have seen already in the examples of Lamport and Krantz, it is possible to express the same using natural language, terms and formulae or both. A theory of semantics construction should be able to identify that 'The greatest common divisor of a and p is 1', and '(a,p) = 1' describe the same. This is different to the formulae '$xa+yp = 1$' and '$xab+ypb = b$' occurring in discourse (1), albeit the fact that term rewriting is semantics preserving.

A first and common kind of reference is connected with the use of terms and formulae in mathematical texts. In the examples (4a–4c), we have expressions that refer to term structures.

(4) a. We obtain

$$p_1^{a_1} \dots p_i^{a_i - b_i} \dots p_k^{a_k} = p_1^{b_1} \dots p_{i-1}^{b_i - 1} p_{i+1}^{b_i + 1} \dots p_k^{b_k}.$$

<u>The left-hand side</u> is divisible by p_i, while <u>the right-hand</u> side is not.

b. We have $m = 2^{b_1} 3^{b_2} \dots p_j^{b_j}$, with <u>every b</u> either 0 or 1. There are just 2^j possible choices of <u>the exponents</u> and so not more than 2^j different values of m.

c. We have

$$2^p = (1+1)^p = 1 + \binom{p}{1} + \dots + \binom{p}{p} = 2 + \sum_{1}^{p-1} \binom{p}{l}.$$

<u>Every term on the right, except the first</u>, is divisible by p.

For resolving the referential expressions 'the left-hand side' and 'the right hand side', apparently, an equation has to be identified, and, respectively, its first or second argument returned.

In the second example, 'every b' quantifies over the set $\{b_1, b_2, \dots, b_j\}$, equally referred to by 'the exponents'. To handle these referential expressions, the term structure as well as domain knowledge has to be taken into account to form a set of terms that is to be returned as value of this referential expression. Apparently, in example (4c) 'every term on the right, except the first' is ambiguous, and requires also advanced grouping mechanism.

2.1.3 Formulae, and means to reference them

In mathematical discourse, we can refer to formulae by proper names. In (1), both 'Theorem 3' and 'Euclid's first theorem' refer to the formulae

$$\forall p \forall a \forall b: prime(p) \wedge div(p, mult(a,b)) \rightarrow div(p,a) \vee div(p,b).$$

To access a sub-formulae of this formula we could use 'the premise', 'the first part of the premise' and various other constructions.

Proposition anaphora are common in mathematical texts. Consider the fundamental theorem of arithmetic and one of its proofs in (5).

> THEOREM 2-2. Every integer $a > 1$ can be represented as a product of one or more primes.
>
> (5) *Proof:* The theorem is true for $a = 2$. Assume it to be true for $2, 3, 4, \ldots, a-1$. If a is prime, we are through. Otherwise a has a divisor different from 1 and a, and we have $a = bc$, with $1 < b < a, 1 < c < a$. The induction hypothesis then implies that
>
> $$b = \prod_{i=1}^{s} p'_i, \quad c = \prod_{i=1}^{t} p''_i,$$
>
> with p'_i, p''_i primes and hence $a = p'_1 p'_2 \ldots p'_s p''_1 \ldots p''_t$.

We have underlined three referential expressions, all of which are proposition anaphora. We need to introduce a discourse referent for each of 'the theorem', 'it', and 'the induction hypothesis', all of which are of propositional type. The propositional type is either induced by the verb phrase 'being true for' – only propositions can be true; or by 'implies' – only propositions can imply other propositions.

Semantics construction is complex. The sentence 'The theorem is true for $a = 2$' means '(It is true that) 2 can be represented as a product of one or more primes' (resolving 'the theorem' to the abstract discourse entity that refers to (the content of) the theorem, and instantiating that universally quantified proposition with 2). Semantics construction for the theorem could yield $\forall a, a > 1, \exists p_1 \ldots \exists p_n, a = p_1 \ldots p_n \wedge \forall i, 1 < p_i \leq a \wedge \mathrm{prime}(p_i)$.

2.1.4 Discourse markers and discourse relations.

The discourse (5) contains quite a lot of clue words (e.g., *if*, *otherwise*, *assume*, *implies*, *hence*), and cue phrases (e.g., *is true for*, *we are through*).

The third proof sentence, starting with *if*, could be read as follows: introduce the assumption *a is prime* and conclude *we are through* (signaling proof termination). However, this sentence has to be embedded into a larger context. In fact, *if* introduces a proof per cases construct, where *a is prime* defines the first case, and where the cue phrase *we are through* terminates this case segment. Obviously, *we are through* does not end the whole proof (there are still some sentences to be processed), so it can only terminate a sub-discourse (this first case). The cue phrase *otherwise* initiates the second case and is a kind of ellipsis which has to be reconstructed to: *a is not prime*. The cue word *and* is not a logical conjunction, but belongs to the cue phrase *and we have* that signals forward reasoning. The last marker of the textbook proof is *hence*, indicating that the subsequent phrase is a logical consequence of the former. Note that the last phrase, $a = p'_1 p'_2 \ldots p'_s p''_1 \ldots p''_t$, terminates the proof because it is the last sentence; and it says that we were able to represent for an arbitrary number a its product of primes representation.

Reconsider the phrase (6a), truncated from discourse (5), being changed into (6b) and (6c).

(6) a. The induction hypothesis then implies that $b = \prod_{i=1}^{s} p'_i$.
b. Obviously $b = \prod_{i=1}^{s} p'_i$.
c. It follows that $b = \prod_{i=1}^{s} p'_i$.

In all these cases, we have a formula, $b = \prod_{i=1}^{s} p'_i$ that is supposed to be a logical consequence of some other statements. However, the exact nature of the consequence relation remains unclear in the sentences (6b–6c). In (6b), we are told that the equation is obviously true. In (6c), we are told that the equation follows from some former statement, which however, is not given. Only in (6a), we are given a justification for $b = \prod_{i=1}^{s} p'_i$. But, the induction hypothesis might only be a necessary, but probably not a sufficient condition for this equation. In fact, in all these three given phrases we have to identify the complete set of premises that allows us to conclude $b = \prod_{i=1}^{s} p'_i$. The task is hard since we cannot expect to find all premises necessary to conclude some statement to be explicitly mentioned in the textbook proof.

It shows that discourse markers help to identify segmentation boundaries, the status of an assertion, and the discourse relation that holds between sentences. To fully determine a discourse relation we need to make explicit or add knowledge that is only implicitly stated or missing.

2.2 Mathematical analysis

We give now a mathematical analysis of discourse (5). The theorem has a form similar to *For every integer $n \geq n_0$, the predicate $P(n)$ holds.* One proof method that can be used to prove a theorem of that form is *mathematical induction*[3]. It consists of three steps:

1. Verify that P holds for n_0.
2. Assume $P(n-1)$ is true.
3. Show that $P(n)$ holds.

However, a variation of mathematical induction, referred to as *generalized induction* is used in the proof under study[4]:

1. Assume $P(i)$ for $n_0 \leq i < n$.
2. Show that $P(n)$ holds.

Knowing the proof method of generalized induction, it is easy to understand the proof. We find the induction hypothesis in the second phrase of the proof. The rest of the proof is the induction step, where $P(n)$ is proved using the fact that $P(i)$ is true for every i between 2 and n. However, it is the first phrase of the informal proof that is hard to understand and misleading. According to the proof method of generalized induction, it cannot be the base case, since the base case is already contained in the induction hypothesis.

Let us write down the proof in more detail. If we break compound phrases into elementary phrases omitting discourse markers (which, nonetheless, are used to interpret the proof), we obtain:

(7) a. The theorem is true for $a = 2$.

b. Assume it to be true for $2, 3, 4, \ldots, a - 1$.

c. a is prime.

d. we are through.

e. a is not prime.

f. a has a divisor different from 1 and a.

g. $a = bc$

h. $1 < b < a$

i. $1 < c < a$

j. $b = \prod_{i=1}^{s} p'_i$, with p'_i primes.

k. $c = \prod_{i=1}^{t} p''_i$, with p''_i primes.

l. $a = p'_1 p'_2 \ldots p'_s p''_1 \ldots p''_t$.

The necessity for the first sentence (7a) will be explained in a more formal account. In (7b), we have the induction hypothesis[5]. The propositions (7c) to (7l) constitute the induction step. Statement (7c) initiates a proof per cases: either a is prime or a is not prime. For (7d), the reasoning is: if a is prime then a is a product of one or more primes, and this concludes the proof for this case. The second case is implicitly indicated in the proof with *otherwise*, which we reconstructed in (7e) to *a is not prime.* The statement in (7f) follows from definitional expansion on the concept of prime using (7e) and negation. In (7g–7i), definitional expansion for the concept of divisor, using (7f), is performed. In (7j–7k), the induction hypothesis (7b) is applied two times, for b using (7h) and for c using (7i). In (7l), simple rewriting takes place using (7g) and (7j–7k).

A more detailed account. The statement to be proven, 'Every integer $a > 1$ can be represented as a product of one or more primes.' has the logical form

(8) $\forall n : P(n)$

The proof is by strong induction. Strong induction can be expression formally by

(9) $\forall n[(\forall k < n : P(k)) \rightarrow P(n)] \rightarrow \forall n : P(n)$

If we can prove

(10) $\forall n[(\forall k < n : P(k)) \rightarrow P(n)]$

then by (9), (10) and Modus Ponens we get (8).
The proof skeleton of (10) is:

let n be arbitrary
  assume $\forall k < n : P(k))$, i.e., $\forall k < n : k > 1 \rightarrow prod\_of\_primes(k)$
  conclude $P(n)$, i.e, $n > 1 \rightarrow prod\_of\_primes(n)$

Since n has been chosen arbitrarily, the proof holds for all n ($\forall$-Introduction).

However, note that for $n = 2$, we have that $\forall k < n[k > 1 \rightarrow prod\_of\_primes(k)]$ in trivially true since there is no k such that $k < n$ and $k > 1$. Therefore, the proof author make a proof per cases.

Case 1 The first case starts with 'The theorem is true for $a = 2$'. Since 2 is prime it is a product of one or more primes.

Case 2 The second case handles $n > 2$. Here, a proof per strong induction is performed as outlined above.

It shows that discourse understanding in the mathematical domain requires a substantial amount of reasoning.

2.3 Requirements for a theory of discourse representation

From the programmer's point of view, a data structure is needed for maintaining entities being introduced and referred to during discourse, and for conditions that must hold for these entities. This data structure need to be structured because mathematical discourse is structured; and sorted because anaphoric expressions can refer to terms (constants, variables, sets of terms), concepts, propositions, and argumentation structures.

From the representational view, the language for discourse representation must

- be able to distinguish between bound and free variables;
- have different kinds of discourse referents
  - for terms[6];
  - for propositions (with modality being assumed, postulated, or derived);
  - for argumentation structures (not being discussed here: e.g., 'the second case', 'the first alternative', 'repeating the argument');
- provide a computational component for accessing and grouping terms and sub-formulae;
- have a naming mechanism;
- have means to describe discourse relations that hold between propositional discourse referents;
- have a deductive component, modeling mathematical reasoning, allowing to compute discourse relations between given propositions, or verifying if a given discourse relations really holds;

We believe that Discourse Representation Theory [10, 20] can be adapted to fit our requirements. In the remainder of the paper, however, we will focus on the use of proof plans that are highly valuable for structuring discourse.

3 The use of proof plans

Our approach is characterized by the use of discourse plans (better: *discourse plan schemas*), representing proof methods, and the fact that we view each (elementary and complex) statement as an abstract discourse entity that can be referred to.

The use of discourse plans is twofold. First, to a large extent, they help to structure the proof. And, discourse plans are indispensable if the text under study lacks discourse markers that explicitly signal structure, or when discourse markers give rise to ambiguity. Second, discourse plans themselves introduce discourse entities: terms, assumptions, goals, and references to sub-proofs which allows us to handle referential expressions like, for example, 'the theorem' and 'the induction hypothesis', we encountered before. Since discourse plans add context, it is straightforward to represent them in the same representation as the discourse.

Generally, proofs have a hierarchical structure. Since proofs have a recursive structure, proof plans consist of sub-plans, and a major step towards discourse understanding is to identify this structure.

In our approach, semantics construction is a process that can be described as follows. It starts with analysing the theorem, and the identification of applicable proof plans. For the theorem of discourse (1), we can apply a proof plan scheme for universally quantified formulae:

(11)
$$\frac{\forall x_1 \ldots \forall x_n : P(x_1, \ldots, x_n)}{\begin{array}{l}\text{LET } \bar{x_1} \ldots \forall \bar{x_n} \text{ be arbitrary} \\ \text{PROVE } P(\bar{x_1}, \ldots, \bar{x_n})\end{array}}$$

Above the line, we have the form of a statement to be proven. Below the line, we have the form of an argumentation structure for proving the statement. The formulae after PROVE defines the remaining proof obligation.

Matching it with the logical form of the theorem of discourse (1) results into:

(12)
$$\frac{\forall p \forall a \forall b : prime(p) \wedge div(p, mult(a,b)) \rightarrow div(p,a) \vee div(p,b)}{\begin{array}{l}\text{LET } \bar{p}, \bar{a}, \bar{b} \text{ be arbitrary} \\ \text{PROVE } prime(\bar{p}) \wedge div(\bar{p}, mult(\bar{a},\bar{b})) \rightarrow div(\bar{p},\bar{a}) \vee div(\bar{p},\bar{b})\end{array}}$$

For the remaining proof obligation we can try either (13a) or (13b), both proofs per elimination.

(13) a.
$$\frac{A \rightarrow B_1 \vee B_2}{\begin{array}{l}\text{ASSUME } A \\ \text{ASSUME } \neg B_1 \\ \text{PROVE } B_2\end{array}}$$

b.
$$\frac{A \rightarrow B_1 \vee B_2}{\begin{array}{l}\text{ASSUME } A \\ \text{ASSUME } \neg B_2 \\ \text{PROVE } B_1\end{array}}$$

As we have demonstrated, theorem analysis defines the context for interpreting the proof. For our example, we get the following proof context:

(14)
$$\frac{\forall p \forall a \forall b : prime(p) \wedge div(p, mult(a,b)) \rightarrow div(p,a) \vee div(p,b)}{\begin{array}{l}\text{LET } \bar{p}, \bar{a}, \bar{b} \text{ be arbitrary} \\ \text{PROVE } prime(\bar{p}) \wedge div(\bar{p}, mult(\bar{a},\bar{b})) \rightarrow div(\bar{p},\bar{a}) \vee div(\bar{p},\bar{b}) \\ \quad \text{ASSUME } prime(\bar{p}) \wedge div(\bar{p}, mult(\bar{a},\bar{b})) \\ \quad \text{ASSUME } \neg div(\bar{p},\bar{a}) \\ \quad \text{PROVE } div(\bar{p},\bar{b})\end{array}}$$

During proof processing other criteria of proof plan applicability will be employed: e.g., referential expressions that refer to abstract discourse entities should be resolvable to abstract discourse entities introduced by a proof plan. This is used to decide between ambiguous proof plan readings.

For processing the proof of discourse (1), we break its complex sentences into elementary assertions, for each of which an abstract discourse referent is introduced. Discourse markers that connected these elementary assertions (e.g., *and*, *therefore*, *assume*) are maintained (attached to them) for further use. Now, we view each of the elementary discourse constituents as anaphoric referring to its place in the proof plan. The discourse markers, we kept them, define constraints for possible sites of constituent attachment. For example, the cue word *hence* indicates that the constituent it introduces, should be attached to the currently "active" proof plan. For the proof of discourse (1), we find that the first two assumptions 'p is prime and $p \mid ab$' (suppose phrase), and '$p \not| a$' (premise of if-then phrase) met the expectations of proof plan (14), and can be attached. The remainder of the proof is now expected to met the proof obligation '$p|b$' (which indeed is the case).
For example, for attaching $(a,p) = 1$, the beginning of the remainder of the proof, we have only one possible attachment point. It is the beginning of the sub-discourse for proving $div(\bar{p}, \bar{b})$.

We get

(15)
$$\frac{\forall p \forall a \forall b : prime(p) \wedge div(p, mult(a,b)) \to div(p,a) \vee div(p,b)}{\begin{array}{l} \text{LET } \bar{p}, \bar{a}, \bar{b} \text{ be arbitrary} \\ \text{PROVE } prime(\bar{p}) \wedge div(\bar{p}, mult(\bar{a},\bar{b})) \to div(\bar{p},\bar{a}) \vee div(\bar{p},\bar{b}) \\ \quad \text{ASSUME } prime(\bar{p}) \wedge div(\bar{p}, mult(\bar{a},\bar{b})) \\ \quad \text{ASSUME } \neg div(\bar{p},\bar{a}) \\ \quad \text{PROVE } div(\bar{p},\bar{b}) \\ \qquad (\bar{a},\bar{p}) = 1 \text{ by } \textit{some justification } \Gamma. \end{array}}$$

The discourse marker 'then' indicates some forward/backward reasoning without introducing structure. We view 'then' as defining a binary discourse relation, $then((\bar{a}, \bar{p}) = 1, \Gamma)$, where Γ can be viewed as an anaphoric entity that refers to all necessary premises which are "logically" necessary for drawing the conclusion $(\bar{a}, \bar{p}) = 1$. For resolving this referential expressions, we have two accessible assumptions (and the 'a', 'p' of '$(a,p) = 1$' resolve to the arbitrarily chosen $\bar{p}$ and $\bar{a}$ introduced by the proof plan).

Discourse markers like *either* or *otherwise* indicate that a new subproof has to be opened and attached. Note that elementary assertions themselves, e.g., the constituent *we are through*, can mark the end of a sub-proof so that consecutive phrases can no longer been attached to this part of the proof which become marked as "closed".

Fig. 2 depicts a proof representation structure for discourse (5). It begs some similarities with the style of proof writing advocated by Lamport [11]. This structure offers a convenient way to refer to a formula. This propositional discourse referents are just the proof line numbers. Also, the representation is structured indicated by indentation and the proof line numbering. It offers an intuitive notion of accessibility for both names (introduced by let) and formulae (either assumed or derived). For example, in proof step **1.2.1.2.2** we can use assumption **1.2.1.2.1** but not assumption **1.2.1.1.1.**. In proof step **1.1.1**, we cannot refer to the name 'b', since it is only introduced later in sub-discourse **1.2.1.2**.

Note that the proof representation structure is still incomplete. That is, for some names introduced we have to decide if they are either bound or free (e.g., the 'b' and 'c' being intro-

LET a be universally quantified
LET k be universally quantified st $k < a$
PROVE $(k > 1 \rightarrow prod\_primes(k)) \rightarrow (a > 1 \rightarrow prod\_primes(a))$

LET $\bar{a}$ be arbitrary

1. *induction_hypothesis*

ASSUME $k > 1 \rightarrow prod\_primes(k)$

1.1. *first_case*

1.1.1. ASSUME $\bar{a} = 2$

1.1.2. QED

1.2. *second_case*

1.2.1. ASSUME $\bar{a} \neq 2$

1.2.1.1. *first_case*

1.2.1.1.1. ASSUME $prime(\bar{a})$

1.2.1.1.2. QED

1.2.1.2. *second_case*

1.2.1.2.1. ASSUME $\neg prime(\bar{a})$

LET X be ?

1.2.1.2.2. $divisor(X, \bar{a})$

LET b, c be ?

1.2.1.2.3. $\bar{a} = bc$

1.2.1.2.4. $1 < b < \bar{a}$

1.2.1.2.5. $1 < c < \bar{a}$

LET $p'_1, p'_2 \ldots, p''_1, p''_2 \ldots$ be ? and prime
LET s, t, i be ?

1.2.1.2.6. $b = \prod_{i=1}^{s} p'_i$

1.2.1.2.7. $c = \prod_{i=1}^{t} p''_i$

1.2.1.2.8. $\bar{a} = p'_1 p'_2 \ldots p'_i p''_1 \ldots p''_t$ (QED)

1.2.2. QED

2. QED

Figure 2: A proof representation structure

duced in sub-discourse **1.2.1.2**). Also, all justifications for derived statements are missing, so that discourse relations are still not computed.

4 Related work

The idea to formally verify textbook proofs has been propagated by McCarthy: *"Checking mathematical proofs is potentially one of the most interesting and useful applications of automatic computer"* [12]. Abrahams, one of his students, wrote a program that initially was intended to analyse textbook proofs. However, as Abrahams noted in [1], this task *"would require far more intelligence than is possible with the current state of the programming art"*. Therefore, so Abrahams, *"the user must create a rigorous, i.e., completely formalized, proof that he believes represents the intent of the author of the textbook proof, and use the computer to check this rigorous proof"*. Abrahams points further out that *"it is a trivial task to program a computer to check a rigorous proof; however, it it not a trivial task to create such a proof from a textbook proof"*. Abrahams was right. In his implementation and in all later projects (e.g., the Mizar project lead by A. Trybulec [14]), proofs had to be written in a *formal language* using a *restricted set of proof construction commands* in order to verify them. A human user is required to fulfill the formalization task.

Similar attempts to parse texts in the mathematics and physics domain are [4] and [5]. Similar to the argument in [5], our purpose is to study natural language understanding in conjunction with automated reasoning. What formal representation can be obtained from textbook proofs and what is needed to formally verify textbook proofs? The most recent work is Simon's PhD thesis [16] which, however, fails to seriously address linguistic issues.

Abstract discourse entities are treated by [3]. Asher differs between fact anaphora, event anaphora, concept anaphora and proposition anaphora. Only the latter has been treated in this paper. The DRS construction process we described begs some similarities to the one described in [3]. A major difference is that, due to our domain, we can assume the existence of discourse plans that establish frames in which discourses must take place.

A more general task than analyzing textbook proofs is to process arbitrary argumentative discourse [6]. Cohen considers a one-way communication where there is a speaker who tries to convince a hearer of some particular argument. The argument understanding system plays the role of the hearer and its task is to analyze the structure of the argument being presented by the speaker. Because argumentative discourse is goal oriented, the discourse has a logical structure. Cohen proposes a three-component model for argument analysis: (i) a theory of expected coherent structure; (ii) a theory of linguistic clue interpretation; and (iii) a theory of evidence relationships. For our domain, (i) is the proof plan; for (ii) we gave linguistic clues; and for (iii) evidence is replaced by establishing a logical relation between propositions of the proof.

An influential theory of discourse structure consisting of the three components *linguistic structure*, *intentional structure* and *attentional state* is developed in [8]. The intentional structure might be matched with the purpose in mathematical discourse to fulfill all proof obligations, whereas the attentional state, in our domain and approach, might coincide with the currently active proof structure being under refinement.

The necessity to perform a multi-level discourse analysis is discussed in [13]. We agree to the fact that a crucial prerequisite to understand the course of the argumentation within a proof is to identify the discourse relations between sentences of that discourse. Deriving the discourse relations of a given textbook proof means reconstructing the intentional structure (describing how sentences within a discourse segment contribute to a common discourse purpose, namely how to decrease the proof obligations) and the informational structure (describing how sentences within a segment are related to each other by some relation, namely a logical consequence relation) of the proof.

In [2], it is argued that a model of discourse plans and their recognition is needed to explain and understand *helpful behaviour*. The problem of discourse understanding can be viewed as a plan recognition problem. The problem of plan recognition is, according to [15], *to take as input a sequence of actions performed by an actor and to infer the goal pursued by the actor and also to organize the action sequence in terms of the plan structure.* For our domain, the actor is the author of the textbook proof, the input is a sequence of sentences, the textbook reader is the plan recognizer.

5 Conclusion and future work

It is argued that mathematical discourse is an ideal domain for studying discourse. We analysed a textbook proof on elementary number theory from both the linguistics and mathematics perspective. It shows, no surprise, that domain knowledge, linguistic knowledge, discourse knowledge, and reasoning are all necessary to successfully build discourse understanders. We proposed to represent proof plans in the same representation as discourse and sketched our approach that describes semantics construction in terms of proof plan recognition and instantiation by matching and attaching. This has to be worked out. Besides the implementation task, future work includes to view discourse representations as first-class objects that can be manipulated in various ways: (i) restructure the proof, i.e., isolate an embedded lemma and its proof and replace it by a reference to that lemma, and (ii) analyse defective discourses and try to repair them.

Notes

[1]cf. Bourbaki, N., *The architecture of mathematics*, In F. Le Lionnais (ed.), Great currents of mathematical thought (Vol. 1), New York: Dover, 1971.

[2]There is a booklet of J. Trzeciak that contains hundreds of standard phrases that allow non-native speakers of English to write mathematical arguments in English [18], enriched with terms and formulae.

[3]See [17], for a introduction to proof methods.

[4]Because in standard mathematical induction, in order to prove $P(n+1)$ we can only refer to $P(n)$. In generalized induction, we can assume that $P(i)$ for every $n_0 < i \leq n$.

[5]Here seems to be a minor flaw. Since it is shown in (7a) that 2 is a product of primes, it is not necessary to assume it in (7b).

[6]To only justify this for constants: in the mathematical domain, a proper name picks out at least one thing, and at most one thing. If we handle constants like proper names in a standard DRT way [10], semantics construction for the sentence '2 is prime' would yield the DRS

x
x = 2 prime(2)

which translates into the logical form $\exists x : x = 2 \wedge prime(x)$. We believe this representational result unsatisfactory for several reasons. Most of all, we believe the logical form unnatural, and the thruth-equivalent $prime(2)$ is the logical form which is to be preferred, because it is much simpler. In addition, to capture the uniqueness of '2', extra presuppositional information has to be supplied.

References

[1] P. W. ABRAHAMS, *Machine verification of mathematical proofs*, PhD thesis, MIT, 1963.

[2] J. ALLEN AND C. PERRAULT, *Analyzing Intention in Utterances*, Artificial Intelligence, 15 (1980), pp. 143–178.

[3] N. ASHER, *Reference to abstract objects in discourse*, Kluwer Academic Publishers, 1993.

[4] D. G. BOBROW, *Natural language input for a computer problem solving system.*, PhD thesis, MIT, 1964.

[5] A. BUNDY, L. BYRD, AND G. LUGER, *Solving mechanics problems using meta-level inference*, in 6th. Int'l Joint Conference on Artificial Intelligence, 1979, pp. 1017–1027.

[6] R. COHEN, *Analyzing the Structure of Argumentative Discourse*, Computational Linguistics, 13 (1987), pp. 11–24.

[7] P. J. DAVIS, *Fidelity in mathematical discourse: Is one and one really two?*, American Mathematical Monthly, 79 (1972).

[8] B. GROSZ AND C. SIDNER, *Attention, Intention, and the Structure of Discourse*, Computational Linguistics, 12 (1986), pp. 175–204.

[9] G. HANNA, *Rigorous proof in mathematics education*, Curriculum Series, The Ontario Institute for Studies in Education, 1983.

[10] H. KAMP AND U. REYLE, *From Discourse to Logic*, Kluwer Academic Publishers, 1993.

[11] L. LAMPORT, *How to write proofs.* `http://www.research.digital.com/SRC/personal/Leslie_Lamport/proofs/proofs.html`, 1993.

[12] J. MCCARTHY, *Computer programs for checking mathematical proofs*, in Recursive Function Theory, Proceedings of Symposia in Pure Mathematics, vol. 5, Americal Mathematical Society, 1962.

[13] J. MOORE AND M. POLLACK, *A Problem for RST: The Need for Multi-Level Discourse Analysis*, Computational Linguistics, 18 (1992), pp. 537–544.

[14] P. RUDNICKI, *An overview of the MIZAR project*, tech. rep., Dept. of Computer Science, University of Alberta, Edmonton, 1992.

[15] C. SCHMIDT, N. SRIDHARAN, AND J. GOODSON, *The Plan Recognition Problem: An Intersection of Psychology and Artificial Intelligence*, Artificial Intelligence, 11 (1978), pp. 45–83.

[16] D. L. SIMON, *Checking Number Theory Proofs in Natural Language*, PhD thesis, UT Austin, 1990.

[17] D. SOLOW, *How to read and do proofs*, John Wiley & Sons, 1990.

[18] J. TRZECIAK, *Writing mathematical papers in english.* Gdańsk Teacher's Press, Institute of Mathematics, Polish Academy of Science, 1993.

[19] S. M. ULAM, *Adventures of a mathematician*, Scibner, New York, 1976.

[20] J. VAN EIJCK AND H. KAMP, *Representing Discourse in Context*, vol. Handbook of Logic & Language, ch. 3, pp. 179–237.

Author's Address:

Claus Zinn
Friedrich-Alexander Universität Erlangen-Nürnberg
Lehrstuhl für Künstliche Intelligenz (IMMD VIII)
Am Weichselgarten 9, D - 91058 Erlangen

mail: `zinn@immd8.informatik.uni-erlangen.de`
www: `http://www8.informatik.uni-erlangen.de/IMMD8/staff/Zinn`