Modal attachment for discourse markers

Jacques Jayez

ENS Lettres et Sciences Humaines

15, Parvis René Descartes

69366 Lyon cedex 07

Jacques.Jayez@ens-lsh.fr

Abstract While it is currently assumed that discourse (in)coherence reflects the (in)existence of an array of connections between discourse segments, the fine structure of these connections or *attachments* is not totally clear. I show here that attachments hold, not strictly between discourse segments, but between the modal objects (information states) which are used by the speech acts issued through the different discourse segments.

1 Introduction

Discourse markers (DMs) are words and expressions which signal that certain discourse relations hold between certain discourse segments, generally sentences, clauses, or sets of them. For instance, *therefore* points to a Consequence or Result discourse relation between segments, while *yet* points to a Contrast or Opposition relation. The number of descriptions of DMs is steadily growing (see Moo segard Hansen (1998), Jayez & Rossari (1999), Jucker & Ziv (1998), Rossari (2000) for some recent surveys). However, no formal toolkit has yet been adopted as a possible standard, in contrast with other areas of semantics (Montague grammar, dynamic semantics and many others). In addition to the intrinsic difficulty of describing the minute differences between DMs in a formal way, I show here that a (partial) formalization has to address the problem of *modal attachment*. Most theories of discourse relations (Mann & Thompson (1985), Sanders et al. (1992), Asher (1993)) assume that a novel discourse segment is *attached* to, that is, connected with, another discourse segment by a discourse relation. Although this works reasonably well for assertions, it cannot be so easily extended to other speech acts (Jayez & Rossari (1998)) or to cases of modal subordination (Kauffman (1997, 2000)). In the next section, using Jayez & Rossari (1998) as a starting point, I explain why the standard view of attachments raises problems with Consequence DMs in French and why the solution proposed by these authors is in part insufficient. In section 3, I present a general dynamic framework to rephrase and solve the problem. Finally, in section 4, I consider the use of nonmonotonic deductive relations to support the strength variations of oppositive DMs.

2 Introducing modal attachments

2.1 The puzzle and a solution

Consider the following examples.

(1) a. Il fait froid, par conséquent ferme la porte (‘It is cold, so close the door’)
b. Sauve le fichier, ??par conséquent fais CTRL-S
(‘Save the file, so type CTRL-S’)
c. Sauve le fichier, fais CTRL-S (‘Save the file, type CTRL-S’)
d. Il a sauvé le fichier, par conséquent il a fait CTR -S
(‘He saved the file, so he typed CTRL-S’)

Par conséquent is a Consequence DM analogous to *so* and *therefore*. Where does the contrast (1b)-(1c) come from? It cannot be due to the imperative since *par conséquent* is compatible with imperatives (see (1a)). It cannot either be due to the fact t hat (1b) is supported by an abductive reasoning of the form ‘if you want to save the file, you need to type CTRL-S’, since *par conséquent* is compatible with abductive reasoning (1d). Finally, it is unclear how a plan-based approach might help us here. To account for the possibility of (1a), we have to assume that the imperative is, in some sense, a consequence of the assertion. This is unproblematic if we associate goals with imperatives, which are then considered as *requests* (Wooldridge 2000, chap. 7).[1] In (1a), the speaker indicates that entertaining the goal of having the door closed is a consequence of the situation (it’s cold). By a similar reasoning, (1b) is understood as mentioning a Consequence relation between two requests. Forbidding this configura tion would be *ad hoc* unless one offers a plan-based motivation for doing so.

Jayez and Rossari (1998) propose that the configuration in (1b) conflicts with general properties of imperatives, when they have the illocutionary force of requests. Normally, imperatives are not felicitous when their propositional content is realized in the situation where the request is issued (see Searle (1969) among others). For instance, one does not consciously request to type CTR -S if this has already been done. Jayez and Rossari attribute the strangeness of (1b) to the fact that the second imperative ‘ type CTRL-S’ is issued in an information state where the propositional content of the first imperative is supposed to hold, that is, where the file has already been saved. Th ey note that this makes sense only if the request of typing CTRL-S is independent of the request of

1 I use *request* as a generic term, which can denote orders, commands and requests proper (that is, moderate demands).

saving the file, but the Consequence DM *par conséquent* connects the two requests, blocking this kind of interpretation.

There are two crucial aspects in this proposal. First, it assumes the existence of request information states, similar to the common ground usually assumed for assertions. Following Stalnaker (1978), many approaches to discourse associate assertions with changes in the shared information state (common ground). Jayez and Rossari extend this idea to requests and I informally summarize their definition of request updates in (2).

(2) **Request update** A request that ϕ by *a* updates the set of worlds representing the future desirable possible worlds according to *a* with ϕ.

The future desirable possible worlds correspond to all the alternatives that an agent rates as possibly true and desirable at some point in the future. The second important facet of the proposal is the role of the DM: most DMs signal discourse relations which have to be evaluated in the 'last' information state, that is, the information state reached through the last update. With Consequence DMs, when a request update takes place, the next information state under consideration is *not* the common ground but the result of the request update. So, there is a notion of modal *localization* behind Jayez and Rossari's account: not only does the content of the various information states change, but the discourse can also lead us to different types of information state. Concerning requests, the proposed solution explains in particular why so-called *pseudo-imperatives* (Clark 1993), like *Close the door and we will be less cold* cannot have a descriptive reading where *we will be less cold* describes a future state of the common ground (in other terms, of the actual world). Since requests are moves to an ideal future information state, the sentence *we will be less cold* is interpreted in this state; it describes what will happen if the door is closed in the ideal state, whence the conditional reading of pseudo-imperatives ('IF you close the door, we will be less cold'). Note that this solution is compatible with plan-based approachs, in particular with Han's (1998) proposal that imperatives that ϕ are instructions to the hearer to add ϕ to her plan-set.

2.2 Problems with the solution

The mentioned solution suffers from two major problems. First, it is couched in the simplest version of propositional dynamic semantics, in the Stalnaker-Heim-Veltman tradition, which does not include pointers to information states. This makes the formulation of constraints formally elusive, even when the intent is intuitively clear. For instance, as there is no multiagent representation, updates such as those define d in (2) are 'anonymous'. We don't know very precisely who is supposed to update which kind of information state. Also, keeping a trace of the different stages of the process associated with a discourse is not possible in the formal language itself. Finall y, the logical properties of the entailment corresponding to the Consequence relation are not integrated smoothly into the approach.

Second, there is no special operator to simulate moves from information state(s) to information state(s). This lack is resp onsible for the impression that the solution is limited to Consequence DMs, while it actually concerns a wide range of discourse markers. In addition, there are minor unwanted complications in the formulation of the update operations. In the next section, I provide a more general and formally clearer solution.

3 A dynamic treatment

3.1 Belief bases

To fix appropriate expressive resources, we need to define information states and operations on them. For Stalnaker and his followers, information states are sets of alternatives (possible worlds). If s is an information state and ϕ a proposition, ϕ is true at s iff it is true in every world $w \in s$. It is possible ('*Might* ϕ is true at s' in Veltman's terminology) iff it is true in at least one world of s. We may adopt a more simple formulation, by extending the notion of *belief base* (Hansson 1998).

(3) **Belief bases** A belief base (BB) is a set of sentences of any suitable first -order modal language.

Complying with the 'philosophy' of BBs, we do not require that they be consistent nor deductively closed. So, a BB can co ntain mutually inconsistent beliefs. Inconsistenc can be removed by revising the BB and is not considered a lethal property, since it can simply express a temporary tension between information sources. More generally, BBs highlight the logical relations between propositions rather than their truth -conditional status in isolation, and they do not necessarily express total information. However, if we admit inconsistent possible worlds or alternatives, a BB can be taken as representing a set of sets of alternatives (that is, a set of information states).

The truth-conditional semantics is standard for non-modal sentences (ϕ is true at s iff ϕ is derivable in s). For modal sentences, we have Kripke models (**B**,**F**), where **B** is a set of BBs and **F** a set of binary accessibility functions. Modal operators are divided into necessity (N-operators) and possibility (P-operators) operators and have truth conditions which reflect the possible inconsistency and partiality of Bbs.

(4) **Truth-conditional semantics for Bbs** Let (**B**,**F**) be a Kripke model. A sentence ϕ is true at $s \in \mathbf{B}$ if: (a) ϕ is non-modal and $\phi \in s$ or $s \vdash \phi$, or (b) ϕ is of the form $N_i\psi$ and ψ is true at $f_i(s)$ or (c) ϕ is of the form $P_i\psi$ and ψ is true at $f_i(s)$ or $\neg\psi$ is not true at $f_i(s)$.

The definition of updates is straightforward.

(5) **Updates** The update of s with ϕ, $s \oplus \phi$, is $s \cup \{\phi\}$.

BBs are ‘anon mous’ as long as we ignore which agent they concern. The aim of multi-agent dynamic semantics (see Gerbrandy (1998) and Baltag (2000) for recent presentations) is precisely to provide a framework where information exchange between agents can be expressed. It would be interesting to incorporate information exchange into the present approach. Unfortunately, space reasons preclude this extension. Using multi-agent models is technically much more complex because, in particular, updates become recursive.

3.1 Basic dynamic tools

Recall that our aim is to provide a formal reflection of moves between BBs. I use a version of First-Order Dynamic Logic (FODL, see Harel et al. (2000)). FODL allows one to have global variables (for BBs) and to distinguish between form ulas and programs. Moves between BBs are programs, which consist of assigning different values to selected global variables. For instance, going from s to s’ can be translated b the sequence $here := s; \pi ; endpt := s'$, where π is some program to be executed and *here* and *endpt* are global variables where current information points (i.e. BBs) are stored. We add to standard FODL the at-operator @, borrowed from hybrid logic (Blackburn & Seligman 1998, Blackburn 2000), which binds the evaluation to a BB: $@_s\phi$ means that ϕ is true at s.

We introduce the reserved variables *here* (for the current point), *sp* (for the starting point), *endpt* (for the endpoint) and *last-assert* for the BB containg the agent's beli efs. Browsing through the syntax, we note that terms are variables for BBs (anonymous variables s, s', etc. or reserved variables), the update result $s \oplus \phi$, the union of two BBs $s \cup s'$, and the result of applying an accessibility function $f_i(s)$. Formulas are those in the first-order modal language plus the at-formulas $@_s\phi$ where ϕ is a formula and s a BB variable. Finally, programs are either assignations $s := t$ (t a term), tests $?\phi$ ('is ϕ true?'), program sequences $\pi ; \pi'$, where π and π' are programs, concu rrent executions $\pi \mid \pi'$ or the null program $\langle\rangle$. $\pi[s \leftarrow s']$ denotes the result of substituting s' for s in π. We now define assignments and the interpretation.

(6) **Assignments** Let *Terms* be the set of terms and *S* the set of possible BBs (the set of sets of sentences of the first -order modal language). An assignment g is a function from *Terms* to *S* such that, in particular:

. $g(s \oplus \phi) = g(s) \cup \phi$,

. $g(s \cup s') = g(s) \cup g(s')$ and

. $g(f_{assert}(s)) = last\text{-}assert$.

Note that the last condition ensu res that f_{assert} always leads us to the BB containing the agent's beliefs.

The interpretation depends on the current BB (s) and assignment pairs (g,h), where $g = h$ when there is no dynamic aspect, such as the jump or addition defined in (8) below.

(7) **Interpretation** If ϕ is a sentence of the first-order modal language, $s,(g,g) \models \phi$ iff ϕ is true at $g(s)$. Otherwise:

1. $s,(g,g) \models @_{s'}\phi$ iff ϕ is true at $g(s')$.
2. $s,(g,h) \models s' := t$ iff $h \approx_{s'} g$ and $h(s') = g(t)$.
3. $s,(g,g) \models ?\phi$ iff $s,(g,g) \models \phi$.
4. $s,(g,h) \models \pi ; \pi'$ iff, for some g', $s,(g,g') \models \pi$ and $s,(g',h) \models \pi'$.

I define two kinds of micro-programs, jumps and additions, whose properties mirror the distinction between adopting a certain perspective and updating an information state.

3.2 Jumps, additions and attachments

(8) **Jumps and additions**

A jump is a program of the form $s := t$. An addition is a program of the form $s := s \oplus \phi$.

The dashed arrow is a jump and the solid arrow an addition. I now turn to the important definitions, which concern the different programs for speech acts, attachments and DMs.

To get a basic understanding of how jumps work, consider assertions; they concern primarily what a given speaker believes. As long as we stay in a BB which contains the speaker's beliefs we ne ed only to update this BB with the content of the successive assertions. Suppose now that a speaker *a* makes a supposition before returning to the BB which contains her ('plain') beliefs.

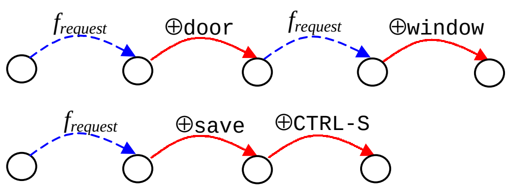
(9) If too many people cannot come, I'll postpone the mee ting. It's a very important meetin

Obviously, the sentence 'it's a very important meeting', is not hypothetical and should not be interpreted in the hypothetical BB where too many people cannot come to the meeting. So, when the first sentence has been pr ocessed, we must be able to 'come back' to the BB which contains *a*'s beliefs. This cannot be done through an accessibilit relation such as f_{belief}, for this relation relates a BB *s* to a BB *s*' only if *s*' represents the information (= what is true or possi ble) contained in the alternatives which are compatible with the information contained in *s*. First, this is not the kind of relation we have between an hypothetical BB and a belief BB. Second, due to the nature of BBs, the f_{belief} function does not make mu ch sense. If the beliefs of *a* are contained in the BB *s*, there is no need to consider another BB which would contain what is plausible to *a* with respect to *s*. This information is already contained in *s*. So we have to use the global variable *last-assert*. Each assertion modifies the value of this variable. Note that 'I'll postpone the meeting' is not a true assertion (that is, not an update of *last-assert*) but either an update of the hypothetical BB where 'too many people cannot come to the meeting' is true or the conclusion of a conditional ('If too many people cannot come, I'll postpone the meeting') with which *last-assert* is updated. In the first case, the hypothetical BB is obtained by going from a BB *s* (*last-assert* in the default case) to its hypothetical counterpart, that is, the set of propositions which (i) are compatible with what is true in *s* and (ii) are not true in *s*.

Generally speaking, determining which BB one must update depends on attachments. The different discourse relations and DMs tell us where to jump before executing additions. For example, the Result discourse relation demands that we use the *endpt* of the speech act to which the next speech act under consideration is attached. It seems that most discourse relations and DMs prefer a jump to *endpt*, but this is not always the case. First, in configurations similar to (9), the nature of some attachments may be unclear. will assume that assertions which have no clear attachment properties simply modif *last-assert*, leaving to more specialize d work the task of discussing the status of discourse relations like Comment (Asher (2000)) or Elaboration (Knott et al. (2001)), which prove rather difficult to detect and define. Second, some discourse relations like Paraphrastic Equivalence (PE) or Elaboration do not give rise to 'modal jumps', in a strong sense.

To see what is at stake, let us consider a simple sequence of commands like 'close the door, then open the window'. It could be represented as (10a). The jump/addition mechanism first moves to the nex *request*-BB, which represents what is possible and desirable with respect to the current BB, it updates with `door` there, then it moves to the nex *request*-BB, from the BB it has just reached and updated and it executes the final addition with `window`. This mirrors the fact that the two actions of closing the door and opening the window are temporally ordered. In contrast, in (1c) ('Save the file, type CTRL-S'). The second request is a precision ('save the file, I mean, type CTR -S') or a paraphrastic reformulation ('Save the file, in other terms, type CTR -S') of the first. The sequence of moves is shown in (10b). We have two additions operating on the *same* BB. In contrast with (10a), the second sentence is not associated with a modal jump through $f_{request}$.

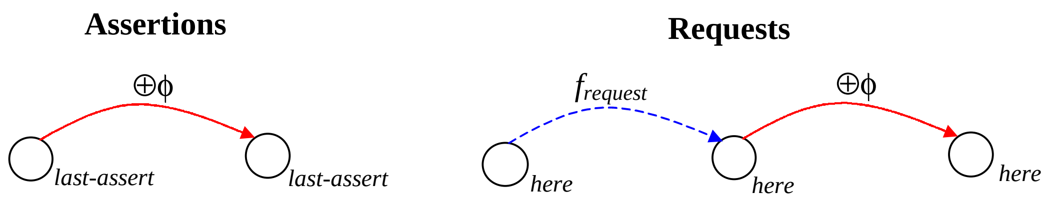
(10) a. b.



The representation in (10b) raises a new question. Normally, requests are avoided when the result they mention is already true. The fact that the result does not already hold is checked when we jump from the current BB s to $f_{request}(s)$. Since it is an intrinsic property of requests, (10b) does not express *two* requests in succession (as (10a) does) but a complex request. I will assume that, in such cases, the two programs are executed concurrently, with the help of the ‘ ’ constructor introduced in the definition of programs.[2] Any program has a starting point (*sp*) where the informational starting point (a BB) is stored. Sequential attachments identify the starting point of the second program to the endpoint (*endpt*) of the first. Concurrent attachments execute both programs starting from the value of *sp* in the first. As they might interact through *here*, this variable must be split up ($here_1$ and $here_2$) to avoid loss of information.

(11) Let Π be a function which transforms speech acts of the form $op(\phi)$ into programs; we have:

1. Assertions, $\Pi(assert(\phi)) = last\text{-}assert := last\text{-}assert \oplus \phi\ ;\ endpt := last\text{-}assert$
2. Requests, $\Pi(request(\phi)) =\ here := sp\ ;\ ?(\neg @_{here}\ \phi)\ ;\ ?(\neg @_{here}\ N_F\phi)\ ;\ here := f_{request}(here)\ ;\ here := here \oplus \phi\ ;\ endpt := here$



Requests activate two tests to check that ϕ is not already true at the current BB and that it won’t be necessarily true in the future (N_F) of the current BB.[3]

As for attachments, we consider only sequential and concurrent ones, ignoring modal subordination and similar phenomena for space reasons. The sequential attachment of two programs π and π’ has the form $\pi\ ;\ sp := endpt\ ;\ \pi'$. The corresponding concurrent

---

[2] An alternative solution is to define programs for requests as, essentially, a sequence (jump/test) + addition, the second request program (‘type CTRL-S’) being then transformed through the Precision, Elaboration or PE attachment. This requires that we perform a little surgery on the corresponding expression. However, it counteracts any compositional analysis in which the programs for speech acts are combined by operators (attachments).

3 In a more precise representation, the operator N_F (i.e. the function f_{NF}) should depend on ϕ and on the context, since the relevant temporal span of the future depends on the propositional content of the request and the interpretation of the speech act.

attachment has the form $\pi[here \leftarrow here_1]$; $\pi'[here \leftarrow here_2]$; $here := here_1 \cup here_2$; $endpt := here$.

3.2 Application to Consequence DMs

One of the ways of capturing the notion of consequence in BBs is to resort Hansson's (1994) kernels. Intuitively, a kernel is any minimal subset which entails a given expression ϕ, where 'minimal' means that suppressing some expression makes impossible to prove ϕ.

(12) **Kernel** The ϕ-kernels of a BB *s* are the minimal subsets of *s* which entail ϕ.[4]

(13) **Relevance** ϕ is relevant to ψ at *s*, in symbols $@_s\ \phi \equiv> \psi$, iff ϕ is a member of at least one ψ-kernel in *s*.

We extend definition (7) by imposing: $s, (g,g) \models \phi \equiv> \psi$ iff ϕ is relevant to ψ in *s*. Note that $\phi \equiv> \psi$ has no meaning in isolation; it expresses logical dependence with respect to a given BB (= a given set of logical resources). Consequence DMs are analyzed as operators on program pairs. They take two programs, corresponding to two speech acts and return a new, more complex, program obtained through the following sequence of actions: (i) the first program (speech act) is executed, (ii) we go to its endpoint, (iii) we check relevance, (iv) we execute the second program. Thus, attachments through Consequence DMs are sequential.

There is no *a priori* constraint upon the attachments through Consequence DMs. Given a sequence $S_1 \ldots S_n$ of discourse segments, a new segment S_{n+1} introduced by a Consequence DM can be attached to any one of the S_i's as long as this choice meets the constraints postulated for attachment. However the combination of st ages (ii) and (iii) might exclude certain possible attachments, as illustrated below. Another point worth of attention is the particular form of relevance checking, which connects a proposition true in a particular BB and a speech act. The intuition here is that the second speech act must be motivated by the *result* of the first.

(14) **Consequence DMs** Let S1 and S2 be two sentences corresponding to the

[4] Kernels are always finite when the notion of proof is finitary.

programs $\Pi(op1(\phi))$ and $\Pi(op2(\psi))$. The interpretation of their attachment through a Consequence DM is the program:

$\Pi(op1(\phi))$; $sp := endpt$; $?(@_{sp}\ \phi \equiv> N_{op2}\psi)$; $\Pi(op2(\psi))$

Let us apply (14) to the examples in (1). We assume that *sp* is an assertive BB and that requests are 'realistic', that is, any desirable possible future world is a possible future world or, in the language of BBs, $g(f_{request}(s)) \subseteq g(f_{PF}(s))$ under any assignment g. The following table gives the different sequences. The light gray regions correspond to the sequential attachments. For (1a), the DM connects the final state and the s tarting point (stage 3) and then checks relevance. The attachment is felt as natural if one can imagine a context in which being cold makes closing the door desirable. For (1b), stage (7) refreshes *sp* which becomes the endpoint of the first program. Since the second speech act must be motivated by the result of the first, we check that saving the file is relevant to the fact that typing CTR -S is desirable and possible. It is unlikely that commonsense situations support this relation: if the file has been s aved, it is no longer necessary to type CTR -S. So, the execution is going to 'fail' at this point. [5] The situation is entirely different for (1c) since there is no DM. It is then possible to attach the second speech act by a concurrent attachment and no implausible commonsense rule has to be appealed to.

(15) **Programs for example (1)**

	(1a)		(1b)		(1c)
1	*last-assert* := *last-assert* ⊕ `cold`	1	*here* : = *sp*	1	$here_1$: = *sp*
2	*endpt* := *last-assert*	2	?(¬@$_{here}$ `save`)	2	?(¬@$_{here_1}$ `save`)
3	*sp* := *endpt*	3	?(¬@$_{here}$ N_F `save`)	3	?(¬@$_{here_1}$ N_F `save`)
4	?(@$_{sp}$ `cold` ≡> $N_{request}$ `door closed`)	4	*here* := $f_{request}$(*here*)	4	$here_1 := f_{request}(here_1)$
5	*here* := *sp*	5	*here* := *here* ⊕ `save`	5	$here_1$:= $here_1$ ⊕ `save`
6	?(¬@$_{here}$ `door closed`)	6	*endpt* := *here*	6	*endpt* := $here_1$
7	?(¬@$_{here}$ N_F `door closed`)	7	*sp* := *endpt*	7	$here_2$: = *sp*
8	*here* := $f_{request}$(*here*)	8	?(@$_{sp}$ `save` ≡> $N_{request}$ `CTRL-S`)	8	?(¬@ $_{here_2}$ `CTRL-S`)
9	*here* := *here* ⊕ `door closed`	9	*here* : = *sp*	9	?(¬@ $_{here_2}$ N_F `CTRL-`

[5] To 'fail' can mean, for instance, to stop or to return an error message which signals that something went wrong.

					S)
10	*endpt* := *here*	10	$?(\neg @_{here}\ \texttt{CTRL-S})$	10	$here_2 := f_{request}(here_2)$
		11	$?(\neg @_{here}\ N_F\ \texttt{CTRL-S})$	11	$here_2 := here_2 \oplus \texttt{CTRL-S}$
		12	$here := f_{request}(here)$	12	$endpt := here_2$
		13	$here := here \oplus \texttt{CTRL-S}$	13	$here := here_1 \cup here_2$
		14	*endpt* := *here*	14	*endpt* := *here*

4 Nonmonotonicity

There are reasons to believe that commonsense reasoning and rhetorical argumentation cannot be reflected by classical modal or non -modal logic (Pollock (1995)). DMs, which are based on informal reasoning, create problems for standard logical approaches (Jayez (1988)). So, the question naturally arises whether the present approach is compatible with a non-standard treatment of logical consequence relations. In the next two sections, I consider briefly the French opposition DM *pourtant* 'yet', studied, for instance, in (Anscombre (1983), Jayez (1981, 1988)) and show how simple non - monotonic techniques can be introduced to address the problems this DM points to.

4.1 Basic properties of pourtant

Pourtant is a *direct oppositive* DM. This means that the propositions it relates contradict each other.

(16) a. Il fait froid, mais j'ai un gros manteau
('It is cold, but I have a thick coat')

b. Il fait froid, ??pourtant j'ai un gros manteau
('It is cold, DM I have thick coat')

As proposed by Anscombre and Ducrot (1977), *mais* can relate two propositions which point to contradictory conclusions. So, (16a) can be interpreted as signalling that 'it is cold' suggests that I will be cold while 'I have a thick coat' suggests rather 'I won't be cold'. *Pourtant* requires that the two propositions 'it is cold' and 'I have a thick coat' be mutually incompatible, which they are not in most contexts. There is another, more intriguing, difference between *pourtant* and *mais*, illustrated by (17).

(17) a. Jean a expliqué franchement la situation à Marie, mais, bien entendu, elle ne l'a pas cru ('Jean candidly explained the situation to Marie, but, of course, she did not believe him')

b. Jean a expliqué franchement la situation à Marie, pourtant, ??bien entendu, elle ne l'a pas cru

(17) suggests that *pourtant* is, in some vague sense, 'stronger' than *mais*. In (17a) and (17b), the direct opposition reading is possible: the fact that Jean spoke very sincerely to Marie creates an expectation that Marie believed him. In this respect, both DMs would be appropriate if the parenthetical *bien entendu* 'of course' were omitted. The data in (16) and (17) show that *pourtant* raises two problems. First, if *pourtant* exploits a direct opposition between two propositions which are independently asserted, how is i t that the speaker does not sound grossly inconsistent when she uses this DM? Second, how must we encode the difference between *pourtant* and *mais* when the discourse features a direct opposition relation

4.2 A Simple nonmonotonic treatment

Concerning the contradiction problem, there are two possible and mutually compatible answers. As BBs are simply sets of propositions (not 'possible worlds'), they can be inconsistent. Inconsistency simply signals that the agent entertains different perspectives, corresponding to different sets of premises, on the same situation. Inconsistency can be removed *via* revision (Hansson (1998)), for instance b suppressing certain premises. Moreover, the consequence relation can be nonmonotonic. Recall that a consequence relation $|\sim$ is nonmonotonic if extending the set of premisses for a given conclusion may suspend the conclusion.

(18) **Nonmonotonicity** $|\sim$ is nonmonotonic if there exists ϕ, Σ, Σ' such that $\Sigma \mid\sim \phi$, $\Sigma \subset \Sigma'$ and $\neg(\Sigma' \mid\sim \phi)$.

Conceptually and formally, nonmonotonicity is a very complex issue which I cannot consider here with enough precision (see Veltman (1996), Antoniou (1997) for some general references). I will give a simple treatment, which requires only minor modifications to the general framework of BBs. Additional expressive power, to distinguish, for instance, between different sorts of exceptions (Veltman (1996)), would require a detailed study of the nonmonotonic relation $|\!\sim$.

When it comes to BBs, the definition of nonmonotonicity must make room for a certain notion of stability.[6] When a proposition ϕ is a nonmonotonic consequence of a BB, it means that (i) some subset of the BB allows for a proof of ϕ and (ii) every subset which allows for a proof of $\neg\phi$ can be embedded in a subset which allows fo r a proof of ϕ. More intuitively, there is no means of definitely proving $\neg\phi$ in the BB.

(19) **Nonmonotonic entailment by a BB** A BB s nonmonotonically entails a proposition ϕ, in symbols $s \mathrel{\|\!\sim} \phi$, iff (i) $\Sigma \mathrel{|\!\sim} \phi$ for some finite $\Sigma \subseteq s$ and (ii) for every finite $\Sigma' \subset s$ such that $\Sigma' \mathrel{|\!\sim} \neg\phi$, there exists a finite Σ'' such that $\Sigma'' \subseteq s$, $\Sigma' \subset \Sigma''$ and $\Sigma'' \mathrel{|\!\sim} \phi$.

Since $|\!\sim$ is nonmonotonic, it may happen that, for some finite Σ' and $\Sigma'' \subset s$, $\Sigma' \mathrel{|\!\sim} \neg\phi$ and $\Sigma'' \mathrel{|\!\sim} \phi$. The semantics of nonmonotonic consequence relati ons is less clear than that of monotonic ones, and I will not try to discuss the different possibilities here. The only requirement I will put on $|\!\sim$ is that it respect completeness with respect to $|=$, that is, if ϕ is a non-modal expression, $s,(g,g) \models \phi$ iff $s \mathrel{\|\!\sim} \phi$ and, if ϕ is a modal sentence of the form $M\psi$, $s,(g,g) \models \phi$ iff, for s' such that $f_M(s) = s'$, we have $s' \mathrel{\|\!\sim} \psi$ (necessity) or $\neg(s' \mathrel{\|\!\sim} \neg\psi)$ (possibility). This means that the BB gives us the resources to deduce the conclusion ϕ when ϕ is a non-modal sentence or that the f_M-BB gives us the resources to deduce ϕ (modal necessity sentences) or does not support the deduction of $\neg\phi$ (modal possibility sentences). The other conditions are similar to those of definition 7. In particular, $s, (g,g) \models @_{s'} \phi$ iff $s' \models \phi$. An important consequence of (19) is (20).

(20) It is not possible to have $s \mathrel{\|\!\sim} \phi$ and $s \mathrel{\|\!\sim} \neg\phi$, for any ϕ.

Suppose that both hold for s. We form an increasing chain of premise sets $\Sigma_0 \mathrel{\|\!\sim} \phi$, $\Sigma_1 \mathrel{\|\!\sim} \neg\phi$ and $\Sigma_0 \subset \Sigma_1$, $\Sigma_2 \mathrel{\|\!\sim} \phi$ and $\Sigma_1 \subset \Sigma_2$, etc. If this chain is finite, then one of the two conditions $s \mathrel{\|\!\sim} \phi$ or $s \mathrel{\|\!\sim} \neg\phi$ fails, contrary to the assumption. If it is infinite, then for ϕ or $\neg\phi$, there is only an infinite set of premises at some stage. So one of the two conditions $s \mathrel{\|\!\sim} \phi$ or $s \mathrel{\|\!\sim} \neg\phi$ fails, contrary to the assumption. Yet, the chain has to be

[6] I won't discuss here the relation of this notion with the 'stable semantics' used in the nonmonotonic approach to logic programming. See Wang (1999) for a recent analysis.

finite or infinite.

Suppose that we have a BB *s* in which explaining frankly the situation to Marie normally entails her believing that the situation holds. Technically, this means that, in *s*, for some finite $\Sigma \subseteq s$, Σ, `frank-Jean` |~ `believe-Marie`. So, we can prove that `believe-Marie` is true with respect to the point of view (= the information resources) corresponding to $\Sigma \cup \{$`frank-Jean`$\}$. Describing the proof-theoretic status of `believe-Marie` in that way is not sufficient, however. *Pourtant* suggests that, surprisingly, Marie did not believe Jean. It is 'strong' in that it implies that the proposition in its scope is strictly incompatible with the proposition conveyed by the other term of the attachment. This is why *pourtant* cannot introduce questions or imperatives.

(21) a. Jean a expliqué franchement la situation à Marie, ??pourtant est-ce qu'elle l'a cru ('Jean candidly explained the situati on to Marie, DM did she believe him?')

b. Jean est très sympathique, ??pourtant ne lui fais pas confiance ('Jean is very nice, DM don't trust him')

(21a) and (21b) sound as illocutionary suicides. The speaker describ es a fact before issuing a question and a request and suggesting, through the use of *pourtant*, that the update of the corresponding BB with the propositional content of the second sentence contradicts the previous fact. Questions and requests are of the re sponsibility of the speaker; so if the dynamic operations they stand for contradict another part of the discourse of the same speaker, this speaker can be considered as responsible for the contradiction. The case of assertions is different, since the speak er can pretend to describes facts which seem to be mutually incompatible but with regard to which she has no responsibility, whence a 'surprise' flavour often associated with *pourtant* (Jayez (1981, 1988)). To encode the radical incompatibility expressed b *pourtant*, we must capture the fact that the speaker has no means to resolve the contradiction, as long as she does not revise old information and/or add new information. This means that the proposition introduced b *pourtant*, `believe-Marie` in our example, should be a strong consequence of `frank-Jean` in *s*.

The property of ψ being a strong consequence of some proposition ϕ is a simple adaptation of the definition (19) that includes the role of ϕ. Recycling definition 13, note $\phi \equiv> \psi$ the fact that, for some finite Σ, we have $\Sigma,\phi \mid\sim \psi$ and, for every Σ' such that $\Sigma' \subset \Sigma \cup \{\phi\}$, $\neg(\Sigma' \mid\sim \psi)$.

(22) **ϕ-nonmonotonic entailment** A BB *s* ϕ-entails ψ iff (i) $\phi \equiv> \psi$ in *s* and (ii) for every finite $\Sigma' \subset s$ such that $\Sigma' \mid\sim \neg\psi$, there exists a finite Σ'' such that $\Sigma'' \subseteq s$, $\Sigma' \subset \Sigma''$, $\phi \in \Sigma''$ and $\Sigma'' \mid\sim \psi$. This property is noted $\phi \mid\equiv> \psi$ (at *s*).

The sense of this definition is to prevent, with the help of ϕ**,** any configuration of the BB in which one might show that $\neg\psi$ definitely holds. We have the following relation between (19) and (22).

(23) If $\phi \mid\equiv> \psi$ at *s*, then $s \mid\sim \psi$.

Pourtant can then be defined in a way analogous to Consequence DMs.

(24) Let S1 and S2 be two sentences corresponding to the programs $\Pi(op1(\phi))$ and $\Pi(op2(\psi))$. The interpretation of their attachment through *pourtant* is the program:
$\Pi(op1(\phi))$; $sp := endpt$; $?(@_{sp}\ \phi \mid\equiv> N_{op2}\neg\psi)$; $\Pi(op2(\psi))$

The difference with the direct oppositive *mais* is given by (25).

(25) The interpretation of the attachment of $\Pi(op1(\phi))$ and $\Pi(op2(\psi))$ through a direct oppositive *mais* is the program:
$\Pi(op1(\phi))$; $sp := endpt$; $?(@_{sp}\ \phi \equiv> N_{op2}\neg\psi)$; $\Pi(op2(\psi))$

Summarizing, *pourtant* and the direct oppositive *mais* differ only by the fact that *pourtant* demands that, at *sp*, ψ be ϕ-entailed ($\mid\equiv>$ vs $\equiv>$). Returning to (17), this means intuitively that, in (17a), the speaker conveys the impression that she may use different sets of premises which lead to ψ or $\neg\psi$ (that is, `believe-Marie` or ¬`believe-Marie`). *Bien entendu* signals that ¬`believe-Marie` is an expected 'strong' (in the sense of (19)) consequence of the current BB, but this is not inconsistent with the fact that there are *some* perspectives (= sets of premises) which support `believe-Marie`. In contrast, with (17b), the speaker conveys the impression that (i) `believe-Marie` is a strong consequence of the current BB (the effect of *pourtant*) and that ¬`believe-Marie` is also an expected strong consequence of the current BB (the effect of *bien entendu*).

Let us now see what the formal counterpart of this intuition is. I define *bien entendu* as an operator which accepts a program (= a speech act), tests that the propositional content of the corresponding speech act is en tailed at the current point and proceeds to execute the program.

(26) The program for *bien entendu*, S, where $\Pi(op(\phi))$ corresponds to S is:
$?(@_{sp}\, N_{op}\phi)\ ;\ \Pi(op(\phi))$

The program corresponding to (17b) is as follows:

Jean a expliqué franchement la situation à Marie	def 11.1 ⇨	*last-assert* := *last-assert* ⊕ `frank-Jean` ; *endpt* := *last-assert* (π1)
elle ne l'a pas cru	def 11.1 ⇨	*last-assert* := *last-assert* ⊕ ¬`believe-Marie` ; *endpt* := *last-assert* (π2)
bien entendu, elle ne l'a pas cru	def 26 ⇨	$?(@_{sp}\, N_{assert}$ ¬`believe-Marie`) ; *last-assert* := *last-assert* ⊕ ¬`believe-Marie` ; *endpt* := *last-assert*
Jean …, pourtant, bien entendu, …	def 24 ⇨	π1 ; *sp* := *endpt* ; $?(@_{sp}$ `frank-Jean` |≡> N_{assert} ¬¬`believe-Marie`) ; $?(@_{sp}\, N_{assert}$ ¬`believe -Marie`) ; π2

Let us consider the two terms $?(@_{sp}\, N_{assert}$ ¬`believe-Marie`) = α, created b *bien entendu* and $?(@_{sp}$ `frank-Jean` |≡> N_{assert} ¬¬`believe-Marie`) = β, created b *pourtant*. By the completeness requirement sketched just after (19), α succeeds iff *sp* ||~ N_{assert} ¬`believe-Marie`, that is iff $f_{assert}(sp)$ ||~ ¬`believe-Marie`. We stipulated in (6) that $g(f_{assert}(s))$ = *last-assert*, but, here, *last-assert* = *sp*, so the condition is equivalent to *sp* ||~ ¬`believe-Marie`. Similarly, β succeeds iff it is true at *sp* that `frank-Jean` |≡> N_{assert} ¬¬`believe-Marie`, which entails, by (23), *sp* ||~ N_{assert} ¬¬`believe-Marie` and *sp* ||~ ¬¬`believe-Marie`. So, we have, conjointly, that *sp* ||~ ¬`believe-Marie` and *sp* ||~ ¬¬`believe-Marie`. However, we know by (20) that this is impossible. *Mais* does not create this problem since it uses ≡> and not |≡>.

5 Conclusion

In this paper, I have argued for a *more fine-grained* representation of attachments for DMs. Specifically, I have proposed to replace the current view of attachments as links between discourse segments by a dynamic conception, where attachments (i) are essentially operations and (ii) can appeal to various logical consequence relations. While the proposed account remained sketchy or insufficient on many points (in particular, I did not offer any multi-agent treatment for dialogues), it illustrates the necessity of filling the gap between discourse-based approaches, which generally resort to a battery of all-purpose discourse relations, and lexical semantics, which tries to capture the individual semantic profiles of the various lexical items or constructions which contribute to discourse coherence.

References

ANSCOMBRE, J.C. 1984. *Pour autant*, *pourtant* et *comment* : à petites causes grands effets. Cahiers de Linguistique Française 5, 37-84.

ANSCOMBRE, J.C. & DUCROT, O. 1977. Deux *mais* en français ? Lingua 43, 23-40.

ANTONIOU, G. 1997. Nonmonotonic Reasoning. Cambridge (MA): MIT Press.

ASHER, N. 1993. Reference to Abstract Objects in Discourse. Dordrecht: Kluwer.

ASHER, N. 2000. Truth conditional discourse semantics for parentheticals. Journal of Semantics 17, 31-50.

BALTAG, A. 2000. The semantics of action reports. Available at http://www.cwi.nl /~abaltag/actionreport.ps.

BLACKBURN, P. 2000. Representation, reasoning and relational structures: A hybrid logic manifesto. Journal of the IGPL 8, 339-365.

BLACKBURN, P. & SELIGMAN, J. 1998. What are hybrid languages? Advances in Modal Logic. Volume I, ed. by M. Kracht. et al. Stanford: CSLI. 41-62.

POLLOCK, J.L. 1995. Cognitive Carpentry. Cambridge (MA): MIT Press.

CLARK, B. 1993. Relevance and ‘pseudo-imperatives’. Linguistics and Philosophy 16, 79-121.

GERBRANDY J. 1998. Bisimulations on Planet Kripke. Ph.D. dissertation. Amsterdam: ILLC.

HAN, C.H. 1998. The Structure and Interpretation of Imperatives: Mood and Force in

Universal Grammar. Ph.D. Dissertation, University of Pennsylvania.
HANSSON, S.O. 1994. Kernel contraction. Journal of Symbolic Logic 59, 845-859.
HANSSON, S.O. 1998. A Textbook of Belief Dynamics. Dordrecht: Kluwer.
HAREL, D., KOZEN, D., TIURYN, J. 2000. Dynamic Logic. Cambridge (MA): MIT Press.
JAYEZ, J. 1981. Etude des rapports entre l'argumentation et certains adverbes français. Ph.D. dissertation. University of Aix-Marseille I.
JAYEZ, J. 1988. L'inférence en langue naturelle. Paris: Hermès.
JAYEZ, J. & ROSSARI, C. 1998. Discourse relations versus discourse marker relations. Proceedings of the ACL'98 workshop on discourse relations and discourse markers, 72-78.
JAYEZ, J. & ROSSARI, C. 1999. Pragmatic connectives as predicates. The case of inferential predicates. Predicative Structures in Natural Language and Lexical Knowledge Bases, ed. by P. Saint-Dizier and E. Viegas. Dordrecht: Kluwer. 285-319.
JUCKER, A.H. & ZIV, Y. (eds) 1998. Discourse Markers. Amsterdam: Benjamins.
KAUFMANN, S. 1997. What's at stack in discourse. Proceedings of the 11 th Amsterdam Colloquium, ed. By P. Dekker, M. Stokhof and Y. Venema. Amsterdam: ILLC. 187-192.
KAUFMANN, S. 2000. Dynamic context management. Formalizing the dynamics of information, ed. by M. Faller, S. Kaufmann and M. Pauly. Stanford: CSLI. 171-188.
KNOTT, A., OBERLANDER, J., O'DONNE LL, M., MELLISH, C. 2001. Beyond elaboration: The interaction of relations and focus in coherent text . Text representation: linguistic and psycholinguistic aspects, ed. by T. Sanders, J. Schilperoord and W. Spooren. Amsterdam: Benjamins. 181-196.
MANN, W. C. & THOMPSON, S.A. 1988. Rhetorical structure theory: Toward a functional theory of text organization. Text 8, 243-281.
MOSEGAARD HANSEN, M.B. 1998. The Function of Discourse Particles. Amsterdam: Benjamins.
ROSSARI, C. 2000. Connecteurs et relations de d iscours: des liens entre cognition et signification. Nancy: Presses Universitaires de Nancy.
SANDERS, T.J.M., SPOOREN, W.P.M., NOORDMAN, L.G.M. 1992. Toward a taxonomy of coherence relations. Discourse Processes 15, 1-35.
SEARLE, J.R. 1969. Speech Acts. Cambridge: Cambridge University Press.
STALNAKER, R. C. 1978. Assertion. Syntax and Semantics 9: Pragmatics, ed. by P. Cole. New York: Academic Press. 315-332.

VELTMAN, F. 1996. Defaults in update semantics. Journal of Philosophical Logic 25, 221-261.

WANG, X. 1999. Negation in Logic and Deductive Databases. Ph.D. dissertation. University of Leeds, School of Computer Studies.

WOOLDRIDGE, M. 2000. Reasoning about Rational Agents. Cambridge (MA): MIT Press.