A Rhetorical Structuring Model for Natural Language Generation in Human-Computer Multi-Party Dialogue

Vladimir Popescu, Jean Caelen

A Rhetorical Structuring Model for Natural Language Generation in Human-Computer Multi-Party Dialogue

Vladimir Popescu, Jean Caelen
Laboratoire d'Informatique de Grenoble,
Grenoble Institute of Technology, France
e-mail: {vladimir.popescu, jean.caelen}@imag.fr

Abstract

Multi-party human-computer dialogue research is still in its infancy. Most of the research in this respect either addresses dialogues between pairs of computers, or performs studies on multi-party human dialogue corpora, in order to better understand this type of interaction. Thus, there are only a few computational models for this type of linguistic interaction and this paper tries to fill this gap. However, only the issue of generating linguistically-appropriate speech turns in multi-party dialogue will be addressed here. For this, a formal framework that accounts for multi-party dialogue situations is developed. Then this model is customized, so that only the point of view of the machine is considered. Finally, several particularly interesting multi-party dialogue situations (for service-oriented systems) are enforced with algorithms for rhetorical structure updating for answer generation, in natural language, and evaluated on concrete multi-party dialogue examples.

1 Introduction

Even if dialogues between a computer and only *one* human partner are studied in a rather mature research field [McTear, 2002] and several commercial applications or systems exist in this respect, the situations where the computer is supposed to get involved in a dialogue with several humans at the same time, are still too little studied in a systematic manner.

Thus, several possibilities exist, towards multi-party human-computer dialogue:

- **multi-session** human-computer dialogue, where the machine gets involved in parallel dialogues with several humans; these dialogues are independent in that the speakers do not interact with each other and do not have access to the dialogues between the machine and the other speakers. This type of interaction is particularly interesting for situations involving concurrent access to a limited set of resources (e.g. meeting room reservation in a company); therefore, in this case there are several *classical* dialogues, on which the computer should maintain a coherent representation. Even if there is not a real multi-party dialogue, there is rather little work worldwide in this respect. For instance, the current state of the art is represented by the PVE ("Portail Vocal pour l'Entreprise") system [Nguyen and Caelen, 2005], [Caelen and Xuereb, 2007]. In this system, multiple sessions are handled, at the dialogue control level, through a game theoretic approach, where machine contribution sequences are evaluated via gains that are dependent, at the same time, on the task context (amount of resources, speakers' roles, etc.) and on the speech acts performed by the speakers.)

- **multi-party** human-computer dialogue, where the machine gets involved in *simultaneous* dialogues with several speakers; as in multi-session dialogue, the machine has to keep a coherent view on the dialogues; yet, there is a major difference in regards to the latter situation: in multi-party interaction, the dialogues are simultaneous, all the speakers being at the same place and having access to *all* speakers' utterances. This is why modeling (and *formalizing*) this type of interaction is particularly difficult. However, since around 2000 there is more and more (substantial) research work in this respect, trying either to study the portability of models designed for traditional dialogues, to multi-party dialogue [Ginzburg *et al.*, 2005], or to analyze multi-party dialogue corpora in order to determine the differences between traditional and multi-party dialogues [Popescu-Belis *et al.*, 2007], or even to give a formal account of particular aspects of multi-party dialogue (such as dialogue control) and concerning only some issues (such as the *shared* context between interlocutors) [Larson *et al.*, 2000].

If traditional dialogue between one human and one computer boiled down to immediate applications in services dealing with user assistance in mitigating certain tasks, such as airplane ticket reservation [McTear, 2002], and even multi-session dialogue begins to reach practical interest in tasks regarding the distribution of a limited set of resources to several users, according to their wishes, multi-party dialogue modeling becomes interesting in tutoring applications, or even in services where the computer has to interact with several speakers at the same time (for instance, in a situation where several clients - a family or a group of friends, ask for a service at the same time). Hence, research in this direction is motivated at the same time from a theoretical standpoint and by practical considerations, when more natural and user-friendly dialogue systems are to be deployed.

In this context, this article addresses several issues in modeling semantic and pragmatic aspects concerning machine utterance generation in service-oriented multi-party dialogues. Thus, we first give a formal account, at a rhetorical level, of multi-party dialogues; then, several particular situations, relevant for service-oriented applications are presented in detail. For these situations, the formal framework is enhanced with a procedural account, instantiated in algorithms for driving the computation and update of the rhetorical structures that specify the dialogues. These rhetorical structures are then used to constrain the choice of appropriate linguistic forms, for a given *communicative intention*, specified in logical form. The algorithms lean on the framework developed by the authors for driving the process of natural language generation in traditional dialogues, with a single human speaker[1].

Thus, in this article one will address the following issues: first, a general framework for situating multi-party dialogues from a rhetorical point of view will be defined; then, two important cases will be depicted, according to the "involvement" of the speaker in conversations. This analysis will then be restricted to the *machine* view on the communication situation, more specifically, only the "involved" view of the machine will be taken into account. This (restricted) model is specified at a procedural level, regarding the discourse structure updating process, when a machine speech

[1]This framework has been presented in several papers, such as [Popescu *et al.*, 2007a, Popescu *et al.*, 2007b, Popescu, 2007].

turn is due to be generated. Hence, this paper proposes an approach to natural language generation in multi-party dialogue, that tries to reuse a formal account designed for traditional, single-user dialogues. Finally, the procedural specification of the model will be demonstrated on typical multi-party dialogue examples.

2 General Framework

2.1 Notations

Throughout this paper a set of notational conventions will be used; these notations are provided below:

- $L_1, ..., L_N, L_\alpha, L_\beta, ...$ ::= human speaker (locutor) of identity i, $i \in \{1, ..., N\}$ or α, β, etc.;
- M ::= the computer (machine);
- v_{ij}, $v_{\alpha\beta}$ ::= conversational "vein" between speakers L_i and L_j (i.e., if there is a dialogue between these two speakers, then it takes place *in* the vein v_{ij})[2]; one has that $v_{ij} \equiv v_{ji}$ for all i and j;
- $SDRS_\alpha^{\beta\gamma}$ ::= segmented discourse representation structure (SDRS), expressed in the framework of Segmented Discourse Representation Theory (SDRT), for the dialogue in vein $v_{\beta\gamma}$, as it is *seen* by speaker L_α;
- $SDRS_{ijk}$ ::= union of all SDRSs seen by speaker L_k on veins $v_{i'j'}$ such that $[i', j'] \subseteq [i, j]$;
- $SDRS_{ij}$ ::= $\bigcup_{\forall k} SDRS_{ijk}$;
- $\pi(\alpha,\beta)$ ::= label of an utterance came from L_α and addressed to L_β;
- $\pi(\alpha, B)$ ::= label of an utterance came from L_α and addressed to the set of speakers $L_\beta : \beta \in B$;
- $t(\pi)$ ::= ordinal index of the utterance labeled "π", in the sequence of utterances produced by π's emitter in the dialogue that this speaker is currently involved in;
- $\mathsf{emitter}(\pi)$ ::= emitter of the utterance labeled "π";
- $\mathsf{equals}(n, m)$ ::= predicate resolving to true if and only if $n = m$ for numeric values, or $n \equiv m$, for other types of atoms;
- $\mathsf{turn}(\pi, \alpha)$ ::= $\mathsf{equals}(\mathsf{emitter}(\pi), L_\alpha) \wedge t(\pi)$;
- $K(\pi)$ ::= logic form expressing the semantics of the utterance labeled "π";
- $SDRS^\alpha(i)$ ::= the discourse sub-structure corresponding only to the contribution of L_α to the i-th speech turn;

[2] There is no relationship between our notion of "vein" and the homonym one in D. Cristea's Veins Theory [Cristea *et al.*, 1998].

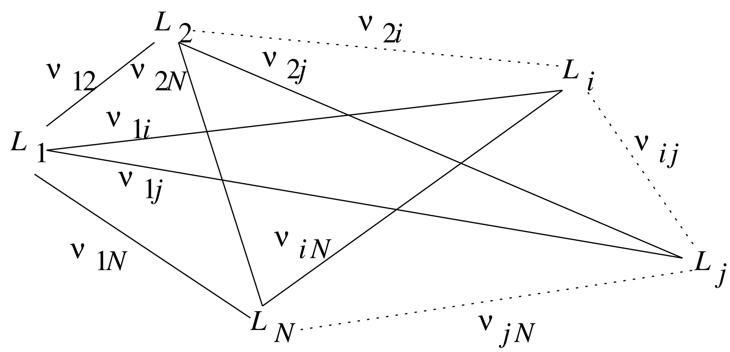

Figure 1: Multi-party dialogue situation

- $\sigma(\alpha,\beta) ::= \{(\alpha,\beta);(\beta,\alpha)\}$, the permutation set of the double (α,β);
- MP ::= discourse "relation" denoting a multi-party dialogue situation; this relation connects the last utterance of one speaker, to the first utterance of another (the subsequent) speaker (except for the machine) in a speech turn;
- $RR^{(c)}_{MON}$ ::= confirmation monologue (third order, see the appendix) rhetorical relation;
- $RR^{(\neg c)}_{MON}$ ::= contradiction monologue rhetorical relation;
- $RR^{(c)}_{DIAL}$ ::= confirmation dialogue (first and second order, see the appendix) rhetorical relation;
- $RR^{(\neg c)}_{DIAL}$ ::= contradiction dialogue rhetorical relation;
- the logical symbols $\exists$, $\forall$, $\Rightarrow$, $\neg$, $\wedge$, $\vee$ have their usual meanings;
- the mathematical symbols $<$, $\leq$, $=$, $\geq$, $>$, $\cup$, $\cap$, $\emptyset$, $\subseteq$, $\subset$, $\supset$, $\supseteq$, $\in$, $\ni$, $\setminus$ have their usual meanings as well.

These notations will allow us to develop the formal account of the *rhetorical view* on multi-party dialogues. This will be done in the following subsection.

2.2 Multi-party Dialogue Situation

For the account given in this section, we assume that there are N speakers in multi-party dialogue situation; this is illustrated in Figure 1, where veins potentially established between (pairs of) speakers are shown.

All the accountable discourse structures for this dialogue situations are specified below:

$$SDRS_{1N} = \bigcup_{i=1}^{N}\left(\bigcup^{N}_{\substack{j=1\\ j\neq i}}\left(SDRS_i^{ij} \cup \left(\bigcup^{N}_{\substack{k=1\\ k\neq j,i}} SDRS_i^{jk}\right)\right)\right)$$

For instance, for three locutors L_i, L_j and L_k, the equation shown above has the following particular form :

$SDRS_{ik} = SDRS_i^{ij} \cup SDRS_i^{ik} \cup SDRS_j^{ji} \cup SDRS_j^{jk} \cup SDRS_k^{ki} \cup SDRS_k^{kj} \cup SDRS_i^{jk} \cup SDRS_j^{ik} \cup SDRS_k^{ij}$.

The formal framework proposed here leans on two postulates:

Postulate 1. *(Common background) Any two speakers involved in a dialogue keep the same view on it. In formal terms:*
$\forall L_i, L_j : \mathit{equals}(SDRS_i^{ij}, SDRS_j^{ij})$

Postulate 2. *(Reflexivity of the dialogue) For any three speakers, two involved in a dialogue and the third one observing them, the view of the latter on the dialogue is invariant with respect to the order of the first two speakers. In formal terms:*
$\forall L_i, L_j, L_k : \mathit{equals}(SDRS_k^{ij}, SDRS_k^{ji})$

These two postulates essentially say that (i) the two speakers in dialogue have the same view on the dialogue they are involved in, and that (ii) for any exterior observer, a dialogue between speaker *A* and speaker *B* is equivalent to the dialogue between speaker *B* and speaker *A*, for the same time span.

Hence, the set of SDRSs that rhetorically represent the multi-party dialogue situation becomes:

$$SDRS_{1N} = \bigcup_{i=1}^{N} \left(\bigcup_{j=i+1}^{N} \left(SDRS_i^{ij} \cup \left(\bigcup^{N}_{\substack{k=1 \\ k \neq j,i}} SDRS_i^{jk} \right) \right) \right)$$

For the particular case of three speakers, the expression becomes:
$SDRS_{ik} = SDRS_i^{ij} \cup SDRS_i^{ik} \cup SDRS_j^{jk} \cup SDRS_i^{jk} \cup SDRS_j^{ik} \cup SDRS_k^{ij}$.

Concerning the "views" that speakers take on the dialogue situation they are involved in, two cases can be depicted:

I. "Involved" view: The speaker participates in the dialogue it accounts for (from a rhetorical perspective, i.e., the corresponding SDRS). We are concerned by discourse structures of the type $SDRS_\alpha^{\alpha\beta}$, for the speaker L_α that builds a representation of the dialogue she undertakes with L_β. Thus, for two couples of speakers $\left(L_\alpha, L_\beta\right)$ and $\left(L_\varphi, L_\psi\right)$, we have two sub-cases:

1. $SDRS_\alpha^{\alpha\beta} \cap SDRS_\varphi^{\varphi\psi} = \emptyset$. In this case we have several dialogues in parallel, which is reducible to the traditional dialogue (involving two speakers); hence, this case will not be given a formal account here;

2. $SDRS_\alpha^{\alpha\beta} \cap SDRS_\varphi^{\varphi\psi} \neq \emptyset$. In this case we have a real multi-party dialogue, hence will be analyzed in this paper. It might be argued that it is possible for a conversation with the same participants and content to take place at two different moments in time, and hence if two conversations intersect with regards of speakers and content, it does not mean that they concern the *same* instance of a conversation. However, this is not the case in our framework, since discourse structures are not persistent from one conversation to another, and hence if two conversations take place at different points in time, there is no relation between them, since, after each dialogue takes place, its traces are not kept.

From the point of view of a speaker involved in a real multi-party dialogue (case I.2. above), if we denote by ρ the label of a rhetorical relation, we can write in an explicit manner the SDRS that L_α builds on the dialogue between herself and a certain L_β:

$$SDRS_\alpha^{\alpha\beta} = \left(\bigcup_{t \text{ speech turn}} (\pi(\sigma(\alpha,\beta)) : \mathsf{equals}(t(\pi), t))\right) \cup \left(\bigcup_{t<t' \text{ speech turns}} (\rho\,(\pi(\sigma(\alpha,\beta)), \pi'(\sigma(\alpha,\beta))) : \mathsf{equals}(t(\pi), t) \wedge \mathsf{equals}(t(\pi'), t'))\right)$$

Thus, a non-void intersection of two discourse structures boils down to the existence of common utterances (that is, utterances that one speaker addresses to several speakers):

$$SDRS_\alpha^{\alpha\beta} \cap SDRS_\varphi^{\varphi\psi} \neq \emptyset \Leftrightarrow \begin{pmatrix} \alpha-\varphi \\ \alpha-\beta \end{pmatrix} \vee \begin{pmatrix} \alpha-\psi \\ \alpha-\beta \end{pmatrix} \vee \begin{pmatrix} \beta-\varphi \\ \beta-\alpha \end{pmatrix} \vee \begin{pmatrix} \beta-\psi \\ \beta-\alpha \end{pmatrix} \vee \begin{pmatrix} \varphi-\alpha \\ \varphi-\psi \end{pmatrix} \vee \begin{pmatrix} \varphi-\beta \\ \varphi-\psi \end{pmatrix} \vee \begin{pmatrix} \psi-\alpha \\ \psi-\varphi \end{pmatrix} \vee \begin{pmatrix} \psi-\beta \\ \psi-\varphi \end{pmatrix}$$

In the latter condensed notation, for three speakers L_X, L_Y, L_Z, the following notation holds:

$$\begin{pmatrix} X-Y \\ X-Z \end{pmatrix} ::= \exists \pi, \pi' : \pi(X,Y) \wedge \pi'(X,Z) \wedge \mathsf{equals}(\mathsf{turn}(\pi, X),\ \mathsf{turn}(\pi', X)) \wedge \mathsf{equals}(K(\pi), K(\pi')).$$

In words, the parenthetic expression to the left means that there exists one utterance produced by a speaker (L_X), that is addressed to two different speakers (L_Y and L_Z).[3]

II. "Exterior" view: The speaker does not participate in the dialogue it accounts for (from a rhetorical point of view), she only *observes* a dialogue between other two speakers. We are concerned with discourse structures of the type $SDRS_\alpha^{\beta\gamma}$. Thus, for two triples of speakers $(L_\alpha, L_\beta, L_\gamma)$ and $(L_\eta, L_\varphi, L_\psi)$ we have two sub-cases:

1. $SDRS_\alpha^{\beta\gamma} \cap SDRS_\eta^{\varphi\psi} = \emptyset$. This case is reducible to simultaneous independent dialogues taking place in parallel, or to multi-session dialogues (if, for instance $\mathsf{equals}(L_\beta, L_\varphi)$); thus, this case will not be accounted for in this paper;

2. $SDRS_\alpha^{\beta\gamma} \cap SDRS_\eta^{\varphi\psi} \neq \emptyset$. This case really represents a multi-party dialogue situation, where two speakers share their views (as observers) on two other dialogues, involving other pairs of speakers. Therefore, this case will be analyzed in detail in this paper.

For speakers observing a pair of real multi-party dialogues (case II. 2 above), we can depict two more sub-cases:

1. $\mathsf{equals}((L_\beta, L_\gamma), (L_\varphi, L_\psi)) \Rightarrow SDRS_\alpha^{\beta\gamma} \cap SDRS_\eta^{\beta\gamma} \neq \emptyset$. This is normal, since there are two different views of the same dialogue (between L_β and L_γ); we have three sub-cases:

   (a) the two "observers" L_α and L_η share the "involved" view of speaker L_β on the dialogue taking place on vein $v_{\beta\gamma}$:
   $SDRS_\alpha^{\beta\gamma} \equiv SDRS_\eta^{\beta\gamma} \equiv SDRS_\beta^{\beta\gamma}$.
   This can happen if L_α and L_η share a lot of background knowledge with L_β and L_γ, being at the same time able to access the whole dialogue that these latter two speakers maintain;

[3] In fact, the meaning of the logic form expressed above is that there exist two utterances labeled π and π', so that π is produced by L_X and addressed to L_Y, and π' is produced by speaker L_X and addressed to L_Z, so that π and π' are produced at the very same time by L_X, and their semantics are identical.

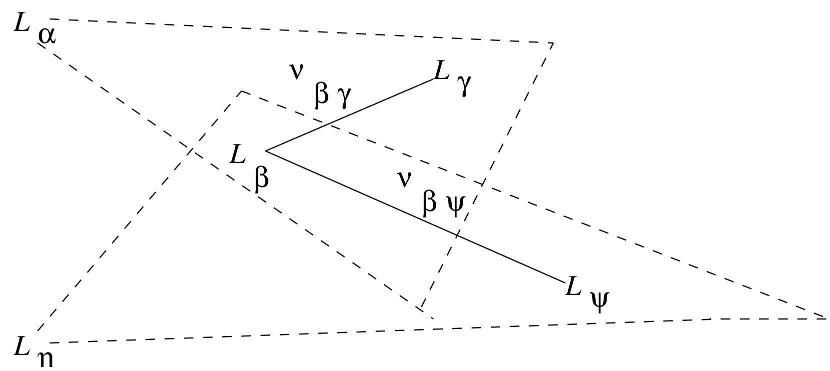

Figure 2: Shared view on speech turns simultaneously addressed to two speakers

(b) the two "observers" L_α and L_η share a partial view, included in L_β's involved view, on the dialogue taking place on vein $\nu_{\beta\gamma}$:
$SDRS_\alpha^{\beta\gamma} \cap SDRS_\eta^{\beta\gamma} \subset SDRS_\beta^{\beta\gamma}$;

(c) the two "observers" L_α and L_η share a view on the dialogue between L_β and L_γ that has nothing to do with the view that L_β and L_γ *share* on this dialogue (by virtue of the common background postulate):
$SDRS_\alpha^{\beta\gamma} \cap SDRS_\eta^{\beta\gamma} \neg \subseteq SDRS_\beta^{\beta\gamma}$. This can happen when L_α and L_η do not have an appropriate background to follow "well" the dialogue taking place on vein $\nu_{\beta\gamma}$;

2. only one of the speakers concerned is the same in both dialogues:
$\mathsf{equals}(L_\beta, L_\varphi) \vee \mathsf{equals}(L_\gamma, L_\psi) \vee \mathsf{equals}(L_\beta, L_\psi) \vee \mathsf{equals}(L_\gamma, L_\varphi)$.
We suppose for instance that $\mathsf{equals}(L_\beta, L_\varphi)$. Thus, case (II. 2) above becomes:
$SDRS_\alpha^{\beta\gamma} \cap SDRS_\eta^{\beta\psi} \neq \emptyset$, that is, L_α and L_η both observe that L_β addresses common turns to L_γ and L_ψ. This situation is depicted in Figure 2.

An interesting sub-case of this latter case presented above is when L_α and L_η share *totally* their views:
$\mathsf{equals}(SDRS_\alpha^{\beta\gamma}, SDRS_\eta^{\beta\gamma}) \wedge \mathsf{equals}(SDRS_\alpha^{\beta\psi}, SDRS_\eta^{\beta\psi})$.

This case is reducible to the *first* sub-case of (II. 2) in order to satisfy the two constraints listed just above; then, once these constraints are satisfied, we have that:
$SDRS_\alpha^{\beta\gamma} \cap SDRS_\eta^{\beta\psi} \neq \emptyset \Leftrightarrow SDRS_\alpha^{\beta\gamma} \cap SDRS_\alpha^{\beta\psi} \neq \emptyset$.
In this case, since we have only one speaker in two different dialogues, we have that:

$$SDRS_\alpha^{\beta\gamma} \cap SDRS_\alpha^{\beta\psi} \neq \emptyset \Leftrightarrow \begin{pmatrix} \beta - \gamma \\ \beta - \psi \end{pmatrix},$$

as seen by speaker L_α.

3 Multi-party Human-Machine Dialogue

3.1 Uniqueness of the Point of View

In Section 2 we had multiple points of view on the dialogues taking place between locutors. In this section, only the point of view of one speaker will be considered, namely machine's point of view.

Thus, assuming that there are N locutors L_1, ..., L_N *and* the machine M, the union of the discourse structures, viewed by the machine, becomes:

$SDRS_{1NM} = \left(\bigcup_{i=1}^{N} SDRS_{M}^{Mi}\right) \cup \left(\bigcup_{i=1}^{N} \left(\bigcup_{j=i+1}^{N} SDRS_{M}^{ij}\right)\right)$.

Moreover, the notations used in Section 2 take a particular form:

$SDRS_{\alpha}^{\alpha\beta} \mapsto SDRS_{M}^{M\beta}$; $SDRS_{\alpha}^{\beta\gamma} \mapsto SDRS_{M}^{\beta\gamma}$.

Therefore, cases I. and II. specified in Section 2 become:

I′. "Involved" view of the machine: For the doubles (M, L_β) and (M, L_φ), we have two cases:

1. $SDRS_{M}^{M\beta} \cap SDRS_{M}^{M\varphi} = \emptyset$. This is a multi-session dialogue situation, where M maintains separate discourse structures for each human interlocutor. This case is not too interesting from a rhetorical point of view, presenting instead serious challenges at a dialogue control level [Nguyen and Caelen, 2005]; an example in this respect, involving the machine M and two speakers L_β and L_φ is shown below (numeric indexes specify the speech turns):

   Session 1:

   M_1: Hello, Sir, what can I do for you?

   L_β^1: Hello, I'd like some DVDs on Kilimanjaro's eruptions...

   M_2: OK... let me see... Here it is: I have found one, on Kilimanjaro's eruption from one hundred thousand years ago. Should I put it on your card?

   L_β^2: Yes, please! Thanks a lot, good bye!

   Session 2:

   M_1: Hello, Sir, what can I do for you?

   L_φ^1: Hi, there! Well, I'd like something on... Kilimanjaro mountain... Can you give me something with pictures or images?

   M_2: Let me see... Well, I have found a touristic leaflet, there are many maps and nice pictures inside...

   L_φ^2: Well, actually I'd like something more like a movie or something... I don't know, do you have a DVD on it?

   M_3: Well, actually we have one, but it has just been given to another customer. It will be available in two weeks...

   L_φ^3: Hum... OK, so can you put in on my reservations list, please?

   M_4: Of course! OK... done! Meanwhile, can I give you something else? For instance, we have another DVD, on volcanoes in Africa, and there is something on Kilimanjaro too!

   L_φ^4: That's perfect, put it on my card right now, please! Thanks, bye!

2. $SDRS_{M}^{M\beta} \cap SDRS_{M}^{M\varphi} \neq \emptyset$. This is a multi-party dialogue, where M is involved in simultaneous dialogues with users L_β and L_φ. A generalization from two speakers (L_β and L_φ), to N speakers (L_1 to L_N) is straightforward. We present below an example of multi-party dialogue illustrating this case, for three speakers, including the machine:

M_1: Hello, what can I do for you?

L_φ^1: Hi, well, we'd like some book or so, on contemporary French theatre...

M_2: Can you please give further specifications, for example on the topic or the author?

L_φ^2: Well, something from the seventeenth century... like Molière or so...

L_β^2: Actually, we'd like something on gamblers... gambler psychology or something...

M_3: So, you would like a seventeenth century play on gambling and gambler psychology, is that right?

L_β^3: Yes, that's right!

L_φ^3: Actually, it needn't be Molière, I mean something lighter will do as well...

M_4: OK, so here it is what I found: "The Player" by Regnard, written in 1696. Would that do?

L_β^4: Well, what's it about?

L_φ^4: Is it moralistic, with a gambler that ends up badly, because of its vice?

M_5: Yes, it is about a gambler, Valère, who trades his mistress's love for gambling, and ends up lonely and broke. Would this do for you?

L_φ^5: Well, this sounds pretty moralistic for me...

L_β^5: And that's precisely what we're looking for: moralistic French seventeenth century theatre on vices such as gambling!

M_6: So, should I put it on your cards? Would a single copy suffice for you, or you would rather like separate copies?

L_β^6: I'd like a separate copy! You can put it on my card!

M_7: OK, so, I put a separate copy for each of you. Is this all right?

L_φ^7: Yes, it's OK, thanks, bye!

L_β^7: Yeah, thanks, bye bye!

In this example, we have a dialogue where L_β and L_φ have a common interest that are trying to convey to the machine, in order for the latter to assist them in achieving their (common as well) goal. Nevertheless, other situations can be imagined, where the two speakers are also negotiating their interests, their goals, and the machine can act as a mediator. However, for the sequences of speech turns where the speakers negotiate their goals, the machine acts as a *listener*, thus adopting an "exterior" view and integrating (the SDRS that represents) this sequence of turns.

In this latter case (I′. 2.), as in Section 2, one can write that:

$$SDRS_M^{M\beta} = \left(\bigcup_{t \text{ speech turn}} (\pi(\sigma(M,\beta)) : \mathsf{equals}(t(\pi), t))\right) \cup \left(\bigcup_{t<t' \text{ speech turns}} (\rho(\pi(\sigma(M,\beta)), \pi'(\sigma(M,\beta))) : \mathsf{equals}(t(\pi), t) \wedge \mathsf{equals}(t(\pi'), t'))\right).$$

Therefore, we have that:

$$SDRS_M^{M\beta} \cap SDRS_M^{M\varphi} \neq \emptyset \Leftrightarrow \begin{pmatrix} M-\beta \\ M-\varphi \end{pmatrix} \vee \begin{pmatrix} \beta - M \\ \beta - \varphi \end{pmatrix} \vee \begin{pmatrix} \varphi - M \\ \varphi - \beta \end{pmatrix}.$$

But:
(i) $\begin{pmatrix} M-\beta \\ M-\varphi \end{pmatrix} \Rightarrow SDRS_M^{M\beta} \cap SDRS_M^{M\varphi} \neq \emptyset$;
(ii) $\begin{pmatrix} \beta-M \\ \beta-\varphi \end{pmatrix} \Rightarrow SDRS_M^{\beta\varphi} \cap SDRS_M^{M\beta} \neq \emptyset$;
(iii) $\begin{pmatrix} \varphi-M \\ \varphi-\beta \end{pmatrix} \Rightarrow SDRS_M^{\beta\varphi} \cap SDRS_M^{M\varphi} \neq \emptyset$.

Cases (ii) and (iii) are included in case (II′), presented below, hence the only situation interesting here is where $\begin{pmatrix} M-\beta \\ M-\varphi \end{pmatrix}$, that is, M produces an utterance addressed at the same time to L_β and L_φ.

II′. "Exterior" view of the machine: For the triples (M, L_β, L_γ) and (M, L_φ, L_ψ), where the computer is only an observer of dialogues between different speakers, we have the two cases:

1. $SDRS_M^{\beta\gamma} \cap SDRS_M^{\varphi\psi} = \emptyset$. In this case M's view on the dialogues taking place on veins $\nu_{\beta\gamma}$ and $\nu_{\varphi\psi}$ do not have anything to do together. In fact, the machine observes two independent dialogues taking place in parallel; this can be useful in tuning M's utterances, addressed to an $L_\lambda \in \{L_\beta, L_\gamma, L_\varphi, L_\psi\}$ (refer to the next section for further details in these regards);

2. $SDRS_M^{\beta\gamma} \cap SDRS_M^{\varphi\psi} \neq \emptyset$. In this case M observes two dialogues having some speech turns in common[4]. Here, there are two more "reasonable" (that is, not in contradiction with common sense) sub-cases:

   (a) $\mathsf{equals}((L_\beta, L_\gamma), (L_\varphi, L_\psi))$. This is a trivial sub-case, since it implies that:
   $SDRS_M^{\beta\gamma} \cap SDRS_M^{\beta\gamma} \neq \emptyset$;

   (b) $\mathsf{equals}(L_\beta, L_\varphi) \vee \mathsf{equals}(L_\gamma, L_\psi) \vee \mathsf{equals}(L_\beta, L_\psi) \vee \mathsf{equals}(L_\gamma, L_\varphi)$. For simplicity, we assume that $\mathsf{equals}(L_\beta, L_\varphi)$, which boils down to:
   $SDRS_M^{\beta\gamma} \cap SDRS_M^{\beta\varphi} \neq \emptyset \Leftrightarrow \begin{pmatrix} \beta-\gamma \\ \beta-\varphi \end{pmatrix}$, as seen by the machine.

Since these latter situations concern dialogues between humans, not directly involving the machine, we will not give examples here.

3.2 Relevant Human-Computer Multi-Party Dialogue Situations

All the multi-party dialogue situations that have been previously presented in this paper have to be detailed further, up to an algorithmic level. Thus, we will specify in the first place the rhetorical structure updating mechanism, when the machine is about to *generate* a speech turn. More specifically, we will have to specify the discourse structure updating mechanism for $\left(SDRS_M^{Mi}\right)_{i=1,\dots,N}$, taking into account $\left(SDRS_M^{ij}\right)_{i\neq j; i,j=1,\dots,N}$.

Thus, we consider two service-oriented dialogue situations, involving the machine and several (N) users (i.e., human locutors):

[4]Please refer to Section 2 in order to see to what stems the fact that two discourse structures have something in common.

A. Service-oriented dialogue whereby the machine (for instance, a librarian or a train ticket seller) talks simultaneously to several clients that don't talk to each other; this dialogue type is appropriate for tutoring contexts as well. In this latter type of interaction, one speaker is the tutor, whereas the other speakers do not talk to anyone else except for the tutor, who in turn takes into account one or several (or all) speakers' knowledge.

Formally, the discourse context as seen by M is:

$SDRS_{1NM} = \bigcup_{i=1}^{N} SDRS_{M}^{Mi}$.

For updating this discourse context, the machine has to produce a speech turn. This turn is first computed in a logic form, by the dialogue controller [Nguyen and Caelen, 2005]:

$\tilde{K}(\pi(M, I)) = \mathsf{Act}(\bigwedge_{i \in I} \mathsf{dest}(L_i) \wedge K(\pi(M, I)))$.

In this equation, Act is the speech act type used to convey the contents of the utterance to be generated [Nguyen and Caelen, 2005], [Popescu *et al.*, 2007b]; the utterance thus specified in logic form is addressed to speakers L_i for $i \in I$, where $I \subseteq \{1, ..., N\}$. The predicate dest/1 specifies that the utterance π is addressed to a speaker that is argument of this predicate.

B. Service-oriented dialogue (as in case (A.) above), whereby the machine talks to a unique user (viz. the tutor or librarian), yet listening to the conversations between this user and the other users involved in the (multi-party) conversation. An example of such a situation is when the machine is a client or a student (for tutoring dialogues) that talks to a unique (main) speaker — e.g. the tutor and, at the same time, listens to the conversations between other speakers and the main speaker. This results in the machine adopting two behaviors: either it listens to conversations between other speakers, or it talks to one (main) speaker. This involves that type **B.** dialogus are subject to the restriction that no overlapping speech turns are produced in conversation; more precisely, that when the machine is talking to the main speaker, the other speakers are not engaged in distinct conversations at the same time. This constraint is quite reasonable in dialogues with a certain degree of formality (e.g. the tutor–students conversations, or even conversation between a group of people and a more distant main speaker).

Formally, the discourse context as seen by M is:

$SDRS_{1NM} = SDRS_{M}^{Mi_0} \cup \left(\bigcup_{j=1; j \neq i_0}^{N} SDRS_{M}^{i_0 j}\right)$.

In this equation, L_{i_0} is the "central" speaker, providing the service (viz. the librarian or tutor).

For updating the discourse context, the machine has to realize in linguistic form a communicative intention (produced in logic form by the dialogue controller) of the type:

$\tilde{K}(\pi(M, i_0)) = \mathsf{Act}(\mathsf{dest}(L_{i_0}) \wedge K(\pi(M, i_0)))$.

Here, the set I specifying the recipients of utterance π is reduced to a singleton, i_0, for the "central" speaker L_{i_0}. The predicate dest/1 making part of the semantic content of $\pi(M, i_0)$ indicates that this utterance is addressed to a certain speaker (L_{i_0}). However, this does not mean that this utterance is actually *heard* by this speaker; it means only that the utterance is *addressed* to this speaker, who might be aware of this or not.

The logic form $\tilde{K}(\pi(M, i_0))$ has to be added to the discourse structure $SDRS_{M}^{Mi0}$, *taking into account* the discourse structures $SDRS_{M}^{i_0 j}$, $j \in \{1,$
$N\} \setminus \{i_0\}$.

These two particular dialogue situations are illustrated in Figure 3.

In case A., the machine is always an "involved" speaker, therefore in situation (I′.), whereas in case B., the machine is either an involved speaker, or about to make the transition from an "observer" speaker to an "involved" one - when the "central" user had been talking to another

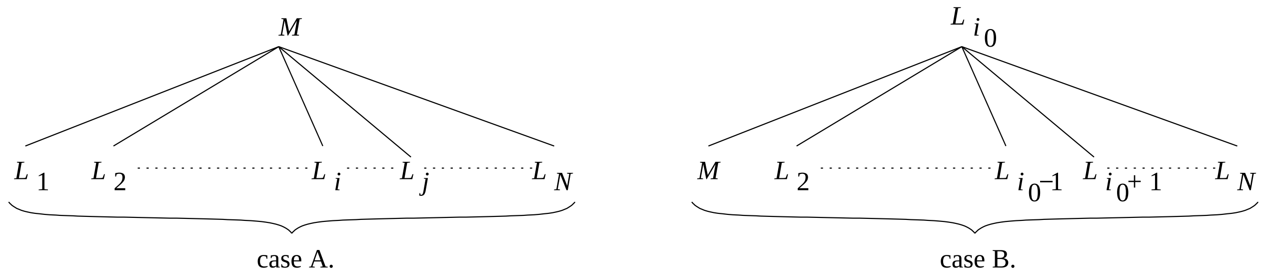

Figure 3: Relevant human-computer multi-party dialogue situations

speaker. Therefore, in this case the machine is either in situation (I′.) as well, or in the course of a transition from situation (II′.) towards situation (I′.).

3.3 Discourse Structure Updating

In order to specify the discourse structure updating process in an algorithmic manner, we lean our attention only on the situation whereby the machine is about to produce a speech turn, which is due to be added to the discourse structures concerned. Moreover, we limit ourselves to the cases A. and B. presented in the previous section. Thus, we have:

Case A. The discourse structures due to be updated are:
$SDRS_{1NM} = \bigcup_{i=1}^{N} SDRS_M^{Mi}$.

The logic form (come from the dialogue controller) to be added to these discourse structures is denoted by $\tilde{K}(\pi(M, I))$ and has the expression presented in the previous subsection.

In this context, the discourse structure updating algorithm is specified below:

for a communicative intention $\tilde{K}(\pi(M, I))$:

1. **find** the set I of recipients of the communicative intention:

   $I = \bigcup_{\mathsf{SubsetOf}(\mathsf{dest}(L_i), \tilde{K}(\pi(M,I)))} \arg(\mathsf{dest}(L_i))$.

   In words, this equation states that the set of the addressess of the communicative intention is determined by grouping all the (distinct) arguments of the dest/1 predicate instantiations in this communicative intention.

2. **choose** the discourse structures to be updated: $\bigcup_{i \in I} SDRS_M^{Mi}$.

   That is, the machine updates the SDRSs of the dialogues that it maintains with the speakers it is talking to (i.e., determined by the set I).

3. **extract** the semantic content from the communicative intention:

   (a) $\mathsf{Act}(\bigwedge_{i \in I} \mathsf{dest}(L_i) \wedge K(\pi(M, I))) \mapsto \mathsf{Act}(\bigwedge_{i \in I} \mathsf{dest}(L_i)) \wedge$
   $\mathsf{Act}(K(\pi(M, I))) \mapsto \bigwedge_{i \in I} \mathsf{Act}(\mathsf{dest}(L_i)) \wedge \mathsf{Act}(K(\pi(M, I)))$;

   (b) $\bigwedge_{i \in I} \mathsf{Act}(\mathsf{dest}(L_i)) \wedge \mathsf{Act}(K(\pi(M, I))) \vee \neg \bigwedge_{i \in I} \mathsf{Act}(\mathsf{dest}(L_i)) \mapsto \bigwedge_{i \in I} (\mathsf{Act}(\mathsf{dest}(L_i)) \vee \neg \mathsf{Act}(\mathsf{dest}(L_i))) \wedge$
   $\mathsf{Act}(K(\pi(M, I))) \mapsto$
   $\mathsf{Act}(K(\pi(M, I)))$;

(c) $\mathsf{Act}(K(\pi(M, I))) \mapsto \mathsf{Act}(K(\pi))$;

In words, this step of the algorithm separates, in the semantic form of the utterance, the part that specifies the addressees of this utterance, from the part that states its propositional content (i.e., literal meaning).

4. **for** $i \in I$: $\mathsf{simple\_update}\left(\mathsf{Act}(K(\pi)), SDRS_M^{Mi}\right)$.

In this algorithm, the function simple_update/2 performs the updating of a discourse structure (given as the second argument), with a logic form, expressing an utterance (given as the first argument), in the case of a traditional dialogue, between two speakers (out of which one is the machine). This procedure, suited for rhetorical structuring in traditional human-computer dialogue, has been extensively described in [Popescu *et al.*, 2007a, Popescu *et al.*, 2007b].

Case B. The discourse structure to be updated is:
$\overline{SDRS}_{1NM} = SDRS_M^{Mi_0}$.

In fact, $SDRS_M^{Mi_0}$ is updated with the semantic content $\mathsf{Act}(K(\pi))$ extracted from the communicative intention $\tilde{K}(\pi(M, i_0))$ by taking into account the discourse structures $\left(SDRS_M^{i_0 j}\right)_{j=1,\ldots,N; j \neq i_0}$. This means that the speech act due to be currently generated is connected to the utterances in the SDRS due to be updated, by taking into account the incidences regarding these utterances. In turn, these incidences are produced through dialogues with the other locutors. Thus, if an utterance in the SDRS concerned
($SDRS_M^{Mi_0}$) is connected, via a "**contradiction**" rhetorical relation (please refer to the Appendix for further details in this respect), to a *subsequent* utterance in another discourse structure (of the type $SDRS_M^{i_0 j}$), then the speech act due to be generated may be connected to the first utterance (in $SDRS_M^{Mi_0}$) in either of the following ways:

- via a contradiction rhetorical relation, with respect to the utterance in $SDRS_M^{i_0 j}$, followed by a **confirmation** rhetorical relation, with respect to the initial utterance in $SDRS_M^{Mi_0}$, by inserting in $SDRS_M^{Mi_0}$ the fragment in $SDRS_M^{i_0 j}$ having realized the contradiction with the first utterance, provided that this utterance had not been committed to by his emitter;
- via an elaboration rhetorical relation (cf. [Asher and Lascarides, 2003], [Popescu *et al.*, 2007a]), with respect to an optional sequence of utterances in structures of the type $SDRS_M^{i_0 j}$, utterances that elaborate in turn on the utterance concerned, in $SDRS_M^{Mi_0}$;
- by default, if there is no constraint induced by discourse structures of the type $SDRS_M^{i_0 j}$, on the utterance concerned, in $SDRS_M^{Mi_0}$.

In view of these elements, a discourse structure updating algorithm is presented below:

for a communicative intention $\tilde{K}(\pi(M, i_0))$:

1. **identify** the discourse structure to be updated: $SDRS_M^{Mi_0}$;
2. **find** the rhetorical structures to be taken into account in the updating process: $\left(SDRS_M^{i_0 j}\right)_{j \notin \{i_0, M\}}$;

3. **for** any utterance π_k already in $SDRS_M^{Mi_0}$:

   (a) **for** any $j \in \{1, ..., N\} \setminus \{i_0\}$:

      i. **check** whether there exists an utterance π'_k not produced by emitter(π_k) in $SDRS_M^{i_0 j}$, so that there exists a rhetorical relation $\rho(\pi_k, \pi'_k)$ in $SDRS_M^{i_0 j}$ as well:

         A. **if** yes (ρ exists), **then**:

         A1) **if** ρ is a contradiction relation, **then**:

            I) **check** whether there exists an utterance π''_k not produced by emitter(π'_k) in $SDRS_M^{i_0 j}$, so that there exists a rhetorical relation $\rho'(\pi'_k, \pi''_k)$ in $SDRS_M^{i_0 j}$:

               I.1) **if** yes, **then**:

                  I.1.1) **if** ρ' is a confirmation rhetorical relation, **then mark** π_k as *non-candidate* utterance for π (which implies that a rhetorical relation between π and π_k will not be calculated);

                  I.1.2) **else**, **keep** π_k in the candidates list for π;

               I.2) **else continue** on step I., iterating over $k'' = \arg(\pi''_k : \pi''_k \in SDRS_M^{i_0 j} \wedge \neg\text{equals}(\text{emitter}(\pi''_k), \text{emitter}(\pi'_k)))$;

            II) **update** the list of non-candidate utterances for $\pi(M, i_0)$, in $SDRS_M^{Mi_0}$:
            $NC(\pi(M, i_0))^{SDRS_M^{Mi_0}} \leftarrow NC(\pi(M, i_0))^{SDRS_M^{Mi_0}}$
            $\cup \{K(\pi_k)\}$;

         A2) **else if** ρ is an elaboration relation (cf. [Asher and Lascarides, 2003], [Popescu *et al.*, 2007a] **then**:

            I) **mark** $K(\pi_k) \wedge K(\pi'_k)$ as a candidate utterance for $\tilde{K}(\pi(M, i_0))$;

            II) **update** the list of *compound* candidates:
            $CC(\pi(M, i_0))^{SDRS_M^{Mi_0} \cup SDRS_M^{i_0 j}} \leftarrow$
            $CC(\pi(M, i_0))^{SDRS_M^{Mi_0} \cup SDRS_M^{i_0 j}} \cup \{\pi_k, K(\pi'_k),$
            $\bigwedge_\rho \rho(\sigma(\pi_k, \pi'_k))\}$;
            in this structure we have the label of utterance π_k, along with the *semantics* of utterance π'_k and with the *labels* of the rhetorical relations connecting these two utterances (the semantics of these rhetorical relations are encoded in the simple_update/2 procedure [Popescu *et al.*, 2007a]);

         A3) **else keep** π_k as a candidate utterance (in the sense pointed out above) for π;

         B. **else** (ρ does not exist), **then** iterate over $k' = \arg(\pi'_k : \pi'_k \in SDRS_M^{i_0 j})$;

      ii. **retrieve** the lists:
      $NC(\pi(M, i_0); j)^{SDRS_M^{Mi_0}} \leftarrow NC(\pi(M, i_0))^{SDRS_M^{Mi_0}}$;
      $CC(\pi(M, i_0); j; k)^{SDRS_M^{Mi_0}} \leftarrow CC(\pi(M, i_0))^{SDRS_M^{Mi_0} \cup SDRS_M^{i_0 j}}$;

   (b) **compute**:

   $NC(\pi(M, i_0)) = \bigcup_j NC(\pi(M, i_0); j)^{SDRS_M^{Mi_0}}$;

   $CC(\pi(M, i_0); k) = \bigcup_j CC(\pi(M, i_0); j; k)^{SDRS_M^{Mi_0}}$;

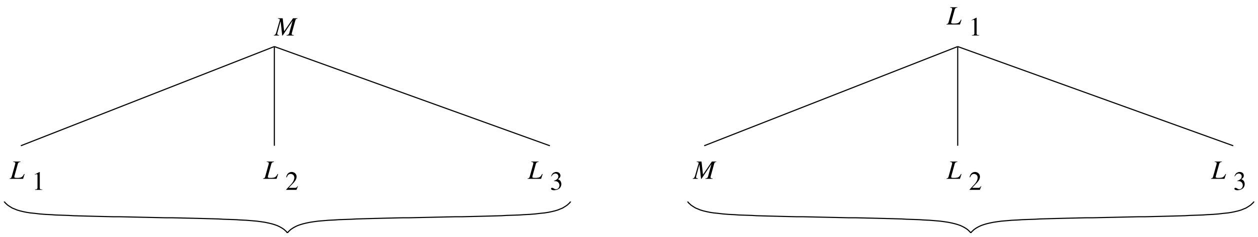

Figure 4: Particular cases ($N = 3$) for the dialogue situations concerned

(c) **update** the discourse structure to be updated, by computing what it remains in the initial SDRS as potential candidates for $\tilde{K}(\pi(M, i_0))$:
$SDRS_M^{Mi_0} \leftarrow SDRS_M^{Mi_0} \setminus NC(\pi(M, i_0))$;

(d) **compute** the *compound* SDRS:
$SDRS_M^{Mi_0}(k) \leftarrow SDRS_M^{Mi_0} \cup CC(\pi(M, i_0); k)$; for any $k = \arg CC(\_; k)$ there exists a π_k in $SDRS_M^{Mi_0}$, since if an utterance had been discarded (as a non-candidate for π), then it would not have been a member in a composition with another SDRS;

4. **compute** the discourse structure to be updated with the utterance due to be generated:
$SDRS_M^{Mi_0} \leftarrow \bigcup_{k:\pi_k \in SDRS_M^{Mi_0}} SDRS_M^{Mi_0}(k)$;

5. extract the semantics for the communicative intention, due to be generated in linguistic form:
$\tilde{K}(\pi(M, i_0)) \mapsto \mathsf{Act}(K(\pi))$;

6. $\mathsf{simple\_update}(\mathsf{Act}(K(\pi)), SDRS_M^{Mi_0})$.

In the two conversation situations A. and B., the main issue concerning multi-party dialogue stems from choosing the discourse structure(s) to update: in situation A., the choice is guided by the set of recipients for the communicative intention due to be generated, whereas in situation B., the choice is driven by the representation that the computer builds on the dialogues that the recipient of the utterance due to be generated establishes with the other speakers. Hence, while in situation A. the choice of the discourse structures is rather obvious and fixed by the dialogue controller, in situation B. the discourse structure to be updated is *built* in an iterative manner, as the dialogue progresses.

4 Multi-Party Human-Computer Dialogue Example

The discourse structure updating mechanism presented in Section 3. will be illustrated for a human-computer dialogue involving three human speakers, in situations A. *and* B. (see Section 3.), when a speech turn is to be produced by the machine. The situation is shown in Figure 4.

For each situation there are several particular cases, according to the addressees of the speech turn due to be produced by the machine:

- Case A.:
  1. $\tilde{K}(\pi(M, I)) \wedge I = \{1, 2, 3\}$;
  2. $\tilde{K}(\pi(M, I)) \wedge I = \{2, 3\}$ (or no matter what double);
  3. $\tilde{K}(\pi(M, I)) \wedge I = \{3\}$ (or no matter what singleton);
- Case B.:
  1. $\tilde{K}(\pi(M, 1)) \wedge \{SDRS_M^{12}, SDRS_M^{13}$ taken into account$\}$;
  2. $\tilde{K}(\pi(M, 1)) \wedge \{SDRS_M^{13}$ taken into account$\}$;
  3. $\tilde{K}(\pi(M, 1)) \wedge \{$ no other discourse structure taken into account$\}$.

Situations A.1, A.2 and A.3 are driven by the dialogue controller (via the predicates "dest/1" in $\tilde{K}(\pi(M, I))$), while situations B.1, B.2 and B.3 are driven by the multi-party dialogue context, in fact by the conversations involving L_1, L_2 and L_3. Thus, a typical human-computer dialogue involving the machine and three human locutors will be analyzed in detail. Here, it will be shown how M can switch automatically from situation A. to situation B. and vice versa, along the same dialogue.

The dialogue concerns a book research and reservation in a library, where the computer is the librarian and L_1, L_2 and L_3 are the clients. It is M that opens the dialogue, introducing itself. The dialogue is annotated in terms of speakers, speech turns and utterances, in the following form:

M_j: $\langle$utterance$\rangle^{\pi_{ji_0}}$...$\langle$utterance$\rangle^{\pi_{ji_1}}$;

L_k^j: $\langle$utterance$\rangle^{\pi_{j(i_1+1)}}$...$\langle$utterance$\rangle^{\pi_{ji_2}}$.

Here, j is the index of the speech turn, k is the index of the human speaker, and π_{js} denotes the s-th utterance in the j-th speech turn.

Thus, the dialogue is listed (in annotated form) below:

M_1: Hello, I am Groplan$^{\pi_{11}}$, I can provide you with assistance in searching for a document in our library.$^{\pi_{12}}$ You can, at any time, ask for information on the availability of a certain book, CD or DVD.$^{\pi_{13}}$ What can I do for you?$^{\pi_{14}}$

L_1^1: Hello$^{\pi_{15}}$, well, we would like a theatre play, on an incest...$^{\pi_{16}}$

L_2^1: An ancient tragedy or something like that...$^{\pi_{17}}$

M_2: Just a minute... here it is:$^{\pi_{21}}$ I found eight ancient tragedies where an incest occurs.$^{\pi_{22}}$ Can you give further specifications, for instance on the author?$^{\pi_{23}}$

L_1^2: Aristophanes?$^{\pi_{24}}$

L_3^2: Or else, Vergilius?$^{\pi_{25}}$

M_3: There is no ancient tragedy written by Aristophanes or Vergilius and involving an incest$^{\pi_{31}}$. Nevertheless, I found three tragedies by Sophocles, one tragedy by Aeschylus and one tragedy by Euripides, where an incest occurs$^{\pi_{32}}$. Would you like to choose one of these authors, find out about other relevant authors, or perform a new search?$^{\pi_{33}}$

L_2^3: What do you have by Sophocles?$^{\pi_{34}}$

M_4: Well, for instance “Oedipus the King”...$^{\pi_{41}}$

L_1^4: What’s this play about?$^{\pi_{42}}$

L_3^4: Is there a brother-sister incest?$^{\pi_{43}}$

M_5: Actually, there is no brother-sister incest in “Oedipus the King”$^{\pi_{51}}$, but it’s essentially about a parricide, followed by a mother-son incest...$^{\pi_{52}}$

L_2^5: Oh yes, this sounds good!$^{\pi_{53}}$

L_1^5: So, it’s like in the movie “My Mother” with Isabelle Huppert.$^{\pi_{54}}$ I loved that movie!$^{\pi_{55}}$

M_6: So, is this tragedy good for you?$^{\pi_{61}}$

L_3^6: Do you have interpretations on DVD as well, for this play?$^{\pi_{62}}$

L_1^6: Oh yeah, an opera, I think that this will do perfectly!$^{\pi_{63}}$

M_7: Yes, indeed, I can give you an opera with Luciano Pavarotti and conducted by Riccardo Muti, played on March 1974 at the theatre Scala of Milan.$^{\pi_{71}}$ Is this OK for you?$^{\pi_{72}}$

L_1^7: Yes, it’s fine for me!$^{\pi_{73}}$

L_3^7: For me, too!$^{\pi_{74}}$

L_2^7: Yes, it’s OK with me!$^{\pi_{75}}$

M_8: So, I’ll lend you “Oedipus the King” at the Hachette Publishers, 1972 edition, and a DVD with the opera with the same title, played at the theatre Scala of Milan, in 1974.$^{\pi_{81}}$ A single copy of each document will do for you$^{\pi_{82}}$, or maybe you’d like a separate copy for each of you?$^{\pi_{83}}$

L_3^8: I would like a separate copy of the book for me.$^{\pi_{84}}$ Still, for the DVD, we’ll watch it together.$^{\pi_{85}}$

L_1^8: The same for me with the DVD$^{\pi_{86}}$, however for the book, it’s OK for me to read it together with my wife (L_2 - n.a.)$^{\pi_{87}}$, so we can have both of us the same copy of the book.$^{\pi_{88}}$

L_2^8: Yes, I agree!$^{\pi_{89}}$

M_9: OK, so, well, I’m lending you one copy for the DVD and two copies for the book.$^{\pi_{91}}$ It this what you want?$^{\pi_{92}}$

L_1^9: Yes, that’s it!$^{\pi_{93}}$

L_3^9: Right!$^{\pi_{94}}$

L_2^9: Yeah, yeah... OK!$^{\pi_{95}}$

M_{10}: OK$^{\pi_{101}}$, so, tell me please, on what name should I make the reservations?$^{\pi_{102}}$

L_1^{10}: Well, I think that we can put a copy of the book on my card, as well as the DVD.$^{\pi_{103}}$

L_3^{10}: And the other copy of the book, you can put it on my card.$^{\pi_{104}}$

M_{11}: OK.$^{\pi_{111}}$ So, here it is...$^{\pi_{112}}$ Thank you for having used our service, see you soon, good bye!$^{\pi_{113}}$ Have a nice day!$^{\pi_{114}}$

L_2^{11}: You have a nice day too!$^{\pi_{115}}$

L_3^{11}: Good bye!$^{\pi_{116}}$

L_1^{11}: Good bye, see you soon!$^{\pi_{117}}$

This dialogue combines the dialogue situations A. and B. specified above, since M, although it remains the "server" (i.e., the librarian), listens to the speech turns whereby the human locutors complete themselves in order to refine their requests. However, the dialogue remains mainly in case A., with fragments where the machine switches to case B. In the subsequent lines we will present a complete trace of the process whereby the discourse structures accounting for this dialogue are updated, with each speech turn produced by the computer.

Yet, before developing the trace on this example, we have to specify the dialogue situation switching process (that is, the way whereby M switches from dialogue situation A. to dialogue situation B. and the other way round). The essential information in this respect resides in the existence or non-existence of MP relations in discourse structures corresponding to multi-party human speech turns. This is formally specified below:

for any speech turn i:
    if $\left(\exists \rho \in SDRS^{(\neg M)}(i) : \mathsf{equals}(\rho, MP)\right)$
        dialogue_situation ← B.;
    else
        dialogue_situation ← A.

In words, the dialogue situation assumed by default by the machine is A., and, if in the current speech turn there is an MP relation in the SDRS accounting for a multi-party dialogue between human users, then the dialogue situation is switched to B. The decision regarding the current dialogue context is made for each speech turn.

These elements being given, the rhetorical structuring of the dialogue shown above takes place as the dialogue progresses:

1. The rhetorical structuring component in the M's natural language generation module places itself by default in dialogue situation A. and builds an SDRS composed of utterances π_{11}, π_{12}, π_{13} and π_{14} and of the rhetorical relations *Elaboration*(π_{11}, π_{12}), *Consequence*(π_{12}, π_{13}) and *Background*(π_{13}, π_{14});

2. The pragmatic interpreter module in M appends human users' utterances π_{15}, π_{16} and π_{17} to the SDRS computed at step 1.; thus, the interpreter first computes *Elaboration*(π_{15}, π_{16}), then it computes *QAP*(π_{14}, *Elaboration*(π_{15}, π_{16})), *QAP*(π_{14}, π_{17}) and finally, given the multi-party dialogue context, $MP(\pi_{16}, \pi_{17})$;

3. The rhetorical structuring component in M's language generator finds the MP relation previously computed, hence it switches to dialogue situation B.; then, the SDRS composed of utterances π_{15} and π_{16}, along with the *Elaboration* relation between them, and utterance π_{17} are selected as *candidate* attachment points for the current M's turn; hence, the machine

first aggregates its communicative intention into utterances π_{21}, π_{22} and π_{23}, then computes the relations *Elaboration*(π_{21}, π_{22}) and *Consequence*(π_{22}, π_{23}); finally, π_{21} is attached to the dialogue history via the relations *P-Elab*(*Elaboration*(π_{15}, π_{16}), π_{21}) and *P-Elab*(π_{17}, π_{21});

4. The pragmatic interpreter first appends π_{24} and π_{25} to the dialogue history via the relations *IQAP*(π_{23}, π_{24}) and *IQAP*(π_{23}, π_{25}); then, it determines the relations $MP(\pi_{24}, \pi_{25})$ and *P-Corr*(π_{24}, π_{25}), hence $MP \wedge P - Corr(\pi_{24}, \pi_{25})$;

5. The rhetorical structuring component in *M*'s generator finds an MP relation between utterances π_{24} and π_{25}, hence it remains in dialogue situation B.; then, it computes the attachment points π_{24} and π_{25}; then, the current *M*'s speech turn is structured, computing the monologue relations *Contrast*(π_{31}, π_{32}) and *Consequence*(π_{32}, π_{33}); then, this turn is appended to the dialogue history via the relations *P-Corr*(π_{24}, π_{31}) and *P-Corr*(π_{25}, π_{31});

6. The pragmatic interpreter in *M* appends user utterance π_{34} to the dialogue history via the relations $Elab_q(\pi_{32}, \pi_{34})$ and *IQAP*(π_{33}, π_{34});

7. The rhetorical structuring component in *M*'s generator, not seeing any MP relation in the previous human speech turn, switches to dialogue situation A., hence, by virtue of the corresponding discourse updating algorithm, appends π_{41} to the dialogue context, via the relation *QAP*(π_{34}, π_{41});

8. The pragmatic interpreter in *M* first computes $MP(\pi_{42}, \pi_{43})$, since $\neg$equals(emitter(π_{42}), emitter(π_{43})), then $Elab_q(\pi_{42}, \pi_{43})$; finally, this sub-structure is appended to the dialogue history via $Background_q(\pi_{41}, \pi_{42})$ and $Elab_q(\pi_{41}, \pi_{43})$;

9. The rhetorical structuring component in *M*'s generator, seeing the MP relation previously computed, switches to dialogue situation B.; then, it aggregates the dialogue intention came from the dialogue controller, into utterances π_{51} and π_{52}, that it rhetorically connects via the relation *Contrast*(π_{51}, π_{52}); then, these utterances are appended to the dialogue history via the relations *P-Corr*(π_{43}, π_{51}) and *QAP*(π_{42}, π_{52});

10. The pragmatic interpreter in *M* first computes $MP(\pi_{53}, \pi_{54})$, then *Elaboration*(π_{54}, π_{55}), then it connects these utterances to the dialogue history, via the relations *ACK*(π_{52}, π_{53}) and *ACK*(π_{52}, *Elaboration*(π_{54}, π_{55}));

11. The rhetorical structuring component in *M*'s generator first sees the MP relation just computed by the pragmatic interpreter, therefore it switches to dialogue situation B., then it connects utterance π_{61} (that corresponds, for the moment, to a communicative intent came from the dialogue controller and expressed in logic form) to the dialogue history: $Elab_q(\pi_{53}, \pi_{61})$ and $Elab_q$(*Elaboration*(π_{54}, π_{55}), π_{61});

12. The pragmatic interpreter in *M* first connects π_{62} to the dialogue history via *Q-Elab*(π_{61}, π_{62}), then computes $MP(\pi_{62}, \pi_{63})$ since these utterances have different emitters; finally, *P-Elab*(π_{62}, π_{63}) is computed as well;

13. The rhetorical structuring component in *M*'s generator first switches to dialogue situation B., since an MP was computed in the previous speech turn, then it aggregates the communicative intention into utterances π_{71} and π_{72}, connected via *Consequence*(π_{71}, π_{72}); finally, this

discourse sub-structure is appended to the dialogue history via *QAP*(π_{62}, π_{71}) and *QAP*(π_{63}, π_{71});

14. The pragmatic interpreter in *M* first computes two *MP* relations between utterances came from the three human users: $MP(\pi_{73}, \pi_{74})$ and $MP(\pi_{74}, \pi_{75})$; then, it connects these utterances to the dialogue history, via the discourse relations *ACK*(π_{72}, π_{73}), *ACK*(π_{72}, π_{74}) and *ACK*(π_{72}, π_{75});

15. The rhetorical structuring component in *M*'s generator first finds the two *MP* relations previously computed, therefore it switches to dialogue situation B. and, by virtue of the corresponding algorithm it establishes the list of possible antecedents in the dialogue history: utterances π_{73}, π_{74} and π_{75}; then, at a monologue level, the machine aggregates the communicative intention into utterances π_{81}, π_{82} and π_{83}, rhetorically connected thus: *Contrast*(π_{82}, π_{83}) and *Elaboration*(π_{81}, *Contrast*(π_{82}, π_{83})); finally, this discourse sub-structure is connected to the possible antecedents in the dialogue history, via the rhetorical relations *P-Elab*(π_{73}, π_{81}), *P-Elab*(π_{74}, π_{81}) and *P-Elab*(π_{75}, π_{81});

16. The pragmatic interpreter in *M* first computes two *MP* relations, $MP(\pi_{85}, \pi_{86})$ and $MP(\pi_{88}, \pi_{89})$, then, it structures the speech turns of each human locutor: *Contrast*(π_{84}, π_{85}) for speaker L_3, *Contrast*(π_{86}, π_{87}) and *Consequence*(π_{87}, π_{88}) for speaker L_1 and π_{89} for speaker L_2; then, the dialogue rhetorical relation *ACK*(π_{88}, π_{89}) is computed; finally, this discourse sub-structure is appended to the dialogue history, via *QAP* (*Contrast*(π_{82}, π_{83}), *Contrast*(π_{84}, π_{85})) and *QAP*(*Contrast*(π_{82}, π_{83}), *Contrast*(π_{86}, π_{87})); we denote the discourse structure computed at this step (and containing utterances π_{84} to π_{89}) by Π;

17. The rhetorical structuring component in *M*'s generator first switches to dialogue situation B. (seeing the *MP* relations computed in the previous step), then aggregates the communicative intention came from the dialogue controller into utterances π_{91} and π_{92} connected via *Consequence*(π_{91}, π_{92}); finally, it appends this sub-structure to the dialogue history via *P-Elab*(Π, π_{91});

18. The pragmatic interpreter in *M* computes two *MP* relations: $MP(\pi_{93}, \pi_{94})$ and $MP(\pi_{94}, \pi_{95})$, then it appends these three utterances to the dialogue history, via *QAP*(π_{92}, π_{93}), *QAP*(π_{92}, π_{94}) and *QAP*(π_{92}, π_{95});

19. The rhetorical structuring component in *M*'s generator first switches to dialogue situation B. (seeing the *MP* relations), then aggregates the communicative intent came from the dialogue controller into utterances π_{101} and π_{102} that it connects via *Elaboration*(π_{101}, π_{102}); finally, it appends this sub-structure to the dialogue history, via *ACK*(π_{93}, π_{101}), *ACK*(π_{94}, π_{101}) and *ACK*(π_{95}, π_{101});

20. The pragmatic interpreter in *M* first computes $MP(\pi_{103}, \pi_{104})$, then it connects these two utterances via *P-Elab*(π_{103}, π_{104}); finally, it appends these utterances to the dialogue context, via *QAP*(π_{102}, π_{103}) and *QAP*(π_{102}, π_{104});

21. The rhetorical structuring component in *M*'s generator first switches to dialogue context B. (seeing the *MP* relation computed in the previous step), then aggregates the communicative

Table 1: Computations performed in the discourse structure updating process

Step no.	M (Pragmatic generation component)	Step no.	$\neg M$ (Pragmatic interpretation component)
1.	(A): $3 \times RR^{(c)}_{MON}$		
		2.	$1 \times RR^{(c)}_{MON} + 1 \times RR^{(c)}_{DIAL} + 1 \times MP$
3.	(B): $2 \times RR^{(c)}_{MON} + 2 \times RR^{(c)}_{DIAL}$		
		4.	$2 \times RR^{(c)}_{DIAL} + 1 \times RR^{(\neg c)}_{DIAL} + 1 \times MP$
5.	(B): $1 \times RR^{(\neg c)}_{MON} + 1 \times RR^{(c)}_{MON} +$ $2 \times RR^{(\neg c)}_{DIAL}$		
		6.	$2 \times RR^{(c)}_{DIAL}$
7.	(A): $1 \times RR^{(c)}_{DIAL}$		
		8.	$3 \times RR^{(c)}_{DIAL} + 1 \times MP$
9.	(B): $1 \times RR^{(\neg c)}_{MON} + 1 \times RR^{(\neg c)}_{DIAL} +$ $1 \times RR^{(c)}_{DIAL}$		
		10.	$1 \times RR^{(c)}_{MON} + 2 \times RR^{(c)}_{DIAL} + 1 \times MP$
11.	(B): $2 \times RR^{(c)}_{DIAL}$		
		12.	$2 \times RR^{(c)}_{DIAL} + 1 \times MP$
13.	(B): $1 \times RR^{(c)}_{MON} + 2 \times RR^{(\neg c)}_{DIAL}$		
		14.	$3 \times RR^{(c)}_{DIAL}$
15.	(B): $1 \times RR^{(\neg c)}_{MON} + 1 \times RR^{(c)}_{MON} +$ $3 \times RR^{(c)}_{DIAL}$		
		16.	$1 \times RR^{(c)}_{MON} + 2 \times RR^{(\neg c)}_{MON} +$ $3 \times RR^{(c)}_{DIAL} + 2 \times MP$
17.	(B): $1 \times RR^{(c)}_{MON} + 1 \times RR^{(c)}_{DIAL}$		
		18.	$3 \times RR^{(c)}_{DIAL} + 2 \times MP$
19.	(B): $1 \times RR^{(c)}_{MON} + 3 \times RR^{(c)}_{DIAL}$		
		20.	$3 \times RR^{(c)}_{DIAL} + 1 \times MP$
21.	(B): $3 \times RR^{(c)}_{MON} + 2 \times RR^{(c)}_{DIAL}$		
		22.	$3 \times RR^{(c)}_{DIAL} + 2 \times MP$

intention came from the dialogue controller, into utterances π_{111}, π_{112}, π_{113} and π_{114}, that it connects via the monologue rhetorical relations *Elaboration*(π_{111}, π_{112}), *Consequence*(π_{112}, π_{113}) and *Elaboration*(π_{113}, π_{114}); finally, it appends this discourse sub-structure to the dialogue context, via $ACK(\pi_{103}, \pi_{111})$ and $ACK(\pi_{104}, \pi_{111})$;

22. The pragmatic interpretation component in M first computes the MP relations $MP(\pi_{115}, \pi_{116})$ and $MP(\pi_{116}, \pi_{117})$, then it appends these utterances to the dialogue history via the dialogue rhetorical relations $ACK(\pi_{114}, \pi_{115})$, $ACK(\pi_{114}, \pi_{116})$ and $ACK(\pi_{114}, \pi_{117})$.

The computations performed in this discourse structure updating process are presented in a concise manner in Table 1, where we denote by $\neg M$ the set of speakers other than M.

For a more intuitive outlook on the trace on the discourse updating process, we present in Figure 5 the discourse structure built for the multi-party dialogue, emphasizing the sub-SDRSs as well.

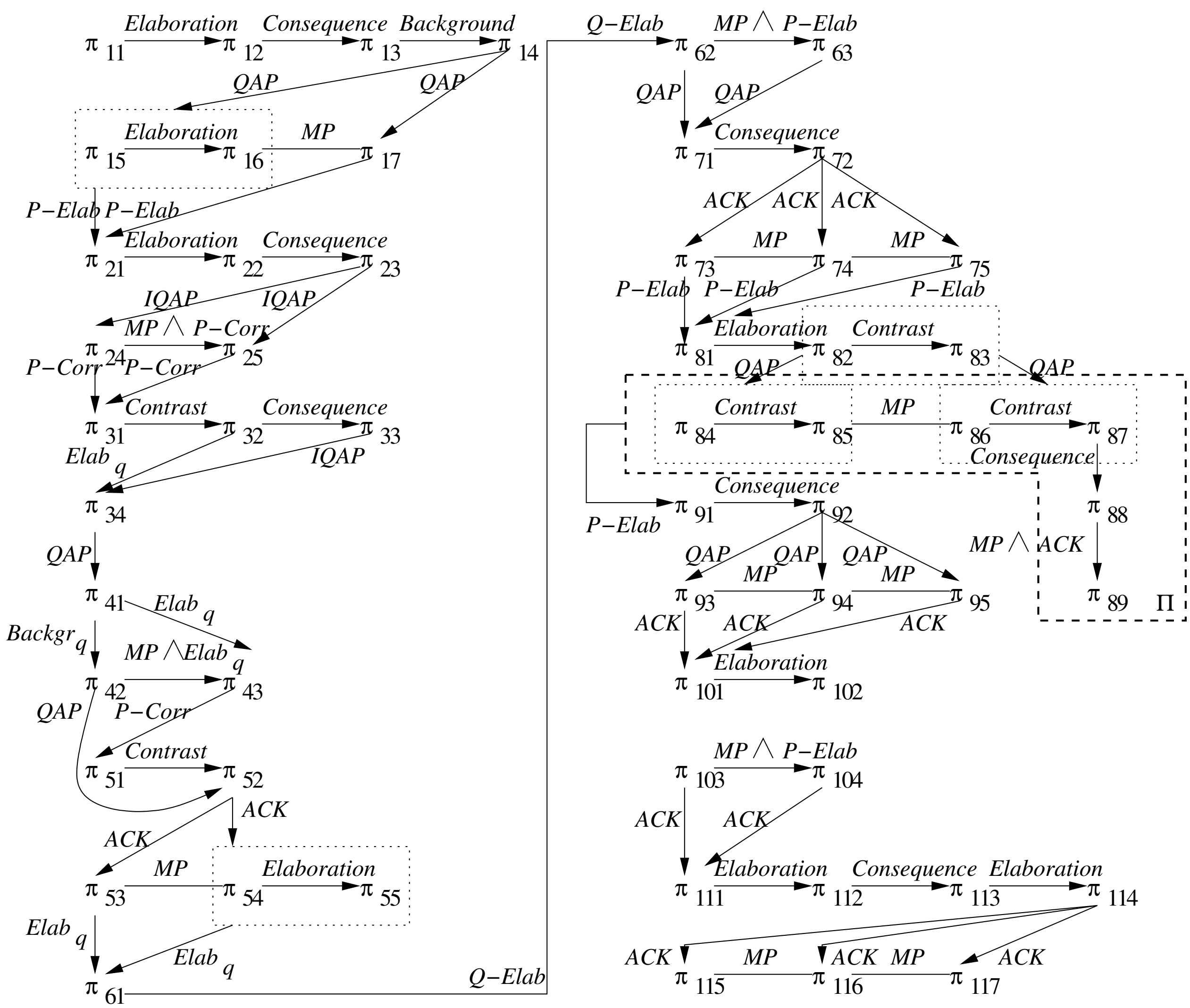

Figure 5: The discourse structure for a multi-party dialogue

5 Discussion and Prospects

In this article we have proposed a formal framework that accounts for multi-party dialogue situations. These specifications are applied to human-computer interaction and illustrated through relevant examples. Thus, although there are limits to the account (such as the inability to handle general multi-party dialogues, i.e., where the interaction is not guided by any "protocol"), the algorithms that lean on this framework, on the one hand, allow the computer to handle more flexible and complex interactions than traditional two-party dialogues, and, on the other hand, give a rather formal and systematic view on real human multi-party dialogues.

However, several issues remain open to research in the near future: (i) the algorithms proposed in this article should be first ran on real multi-party human dialogues (for instance, on theatrical plays where appropriate task ontologies would have been built) and validated on them, (ii) then, these methods should be integrated in a multi-party dialogue architecture (such as extensions performed to TRINDI Kit), in order to drive the answer generation component, (iii) after this step, all the multi-party dialogue situations described in the article should be instantiated in procedural descriptions and then applied to real multi-party dialogues.

In the longer term, the framework presented in this article might form the basis for more complex applications, involving at the same time interactions between artificial agents and humans, in computer games, or applications in the more recent strand related to interactive storytelling [Cavazza *et al.*, 2002], where one (or several) human subject(s) interacts with several artificial agents in a multimodal manner, including natural language. Thus, in this type of applications, the way whereby interacting locutors exchange speech turns is particularly relevant for an entertaining and natural interaction [Thue *et al.*, 2007].

Last but not least, the formal framework proposed in this paper would be useful in driving decisions regarding the fine-tuning of the utterances to be produced by the machine in multi-party dialogue, such as the pronominal anaphora generation process (a mechanism in this respect, concerning only two-party human-computer dialogues, is presented in [Popescu, 2007]), in order to render the utterances generated more relevant to the dialogue situation and to the addresses.

Appendix: A Fragment of SDRT for Language Generation in Human-Computer Dialogue

Even if SDRT is a rather mature formal account of the rhetorical structure of discourse, it remains a theory of discourse *interpretation*. Thus, in order to lean on SDRT for generation purposes, several adaptation need to be performed. First of all, the communicative goal "behind" an utterance (called "Speech Act Related Goal" - SARG in SDRT) is given as input for generation (since it is computed by the dialogue controller), whereas in interpretation this is a big issue to be carried out by the theory [Asher and Lascarides, 2003]. On the contrary, in generation, the big issue, from a rhetorical structuring perspective, resides in stating a way whereby the (already known) SARG constrains its rhetorical potential, that is, the set of rhetorical relations that connect it to previous utterances in the discourse [Popescu *et al.*, 2007a].

Thus, out of the around 35 rhetorical relations proposed in vanilla SDRT (in the 2003 version [Asher and Lascarides, 2003]), a subset of 17 have been chosen. These rhetorical relations, considered particularly useful for human-computer dialogue purposes, are clustered in three types:

- **first-order** rhetorical relations - *Q-Elab*, *IQAP*, *P-Corr* and *P-Elab*, with informal semantics as in [Asher and Lascarides, 2003], that are strongly related to *temporal* aspects in dialogue, hence used in an approximate manner, specific to the type of dialogue concerned (i.e., conversations involving negotiations on time intervals of resource availability, as in Verbmobil corpus [Schlangen *et al.*, 2001]);
- **second-order** rhetorical relations - $Background_q$, $Elab_q$, $Narration_q$, *QAP*, *ACK* and *NEI*, with informal semantics as in [Asher and Lascarides, 2003], that are less constrained by the temporal aspects of the dialogues concerned, hence used in a manner closer to that specified in vanilla SDRT;
- **third-order** rhetorical relations, specific to monologues and used to relate utterances within a speech turn, generated by one of the speakers (either the human or the machine) - *Alternation*, *Background*, *Consequence*, *Elaboration*, *Narration*, *Contrast* and *Parallel*, with semantics as in vanilla SDRT [Asher and Lascarides, 2003].

Furthermore, in order to enhance the semantics of these rhetorical relations with pragmatic aspects, we group them in two categories:

- **confirmation** rhetorical relations - *Q-Elab*, *P-Elab*, $Elab_q$, $Narration_q$, *QAP*, *ACK* and *IQAP*, *Background*, *Consequence*, *Elab*, *Narration* and *Parallel*; via these rhetorical relations the current utterance (appearing a the second argument in these relations) does not question the previous utterance (appearing as the first argument in the relations);
- **contradiction** rhetorical relations - *P-Corr*, $Background_q$, *NEI*, *Alternation* and *Contrast*.

We illustrate below, via appropriate human-computer interaction examples, each rhetorical relation, emphasizing its type, category and informal semantics; here, *U* and *M* designate a human user and a machine, respectively, whereas π_i, designate utterances:

1. *Q-Elab*(π_1, π_2) ("Question Elaboration"):
   - Type: first order;
   - Category: confirmation;
   - Informal semantics: π_2 is a question to which any answer elaborates a plan for achieving the SARG conveyed by π_1;
   - Example:
     π_1: *U*: I will read this book on Monday.
     π_2: *M*: Is it OK for you at 2 o'clock PM?
2. *IQAP*(π_1, π_2) ("Indirect Question-Answer Pair"):
   - Type: first order;
   - Category: confirmation;
   - Informal semantics: π_1 is a question and π_2 is an answer that provides information allowing π_1's emitter to *infer* an answer to π_1;
   - Example:
     π_1: *U*: Could I have this book for next week?
     π_2: *M*: The book is due to be returned back to the library this Wednesday.
3. *P-Corr* (π_1, π_2) ("Plan Correction"):
   - Type: first-order;
   - Category: contradiction;
   - Informal semantics: π_2's emitter refutes the SARG conveyed by π_1;
   - Example:
     π_1: *U*: Could I have this book for next week?
     π_2: *M*: The book is already reserved by another customer since the 16th until the 25th.
4. *P-Elab* (π_1, π_2) ("Plan Elaboration"):
   - Type: first order;
   - Category: confirmation;
   - Informal semantics: π_2 elaborates a plan for achieving the SARG conveyed by π_1;
   - Example:
     π_1: *U*: Could I have this book for next week?
     π_2: *M*: To have it, you have to go at our headquarters Street 'X', to reserve it there, in the beginning of next week.

5. *Background$_q$* (π_1, π_2):

- Type: second order;
- Category: contradiction;
- Informal semantics: π_2 is a question to which any answer is in a *Background* relation to π_1;
- Example:
  π_1: *M*: This book has already been reserved by another customer.
  π_2: *U*: Are there other clients having looked for this book as well?

6. *Elaboration$_q$* (π_1, π_2):

- Type: second order;
- Category: confirmation;
- Informal semantics: akin to that of *Background$_q$*;
- Example:
  π_1: *M*: We have received new books on your field of interest.
  π_2: *U*: Could you give me some titles, please?

7. *Narration$_q$*(π_1, π_2):

- Type: second order;
- Category: confirmation;
- Informal semantics: akin to that of *Background$_q$*;
- Example:
  π_1: *M*: You can start with book 'X'.
  π_2: *U*: And then, what do you recommend me?

8. *QAP*(π_1, π_2) ("Question-Answer Pair"):

- Type: second order;
- Category: confirmation;
- Informal semantics: π_1 is a question and π_2 is a direct answer to this question;
- Example:
  π_1: *U*: Where can I find the book with registration number 'xyz'?
  π_2: *M*: The second floor, to the right.

9. *ACK*(π_1, π_2) ("Acknowledgment"):

- Type: second order;
- Category: confirmation;
- Informal semantics: π_1 is an utterance and π_2 is another utterance, produced by another speaker that π_1's emitter, whereby π_1 is confirmed;
- Example:
  π_1: *M*: Are these books OK for you?
  π_2: *U*: Yes.

10. *NEI*(π_1, π_2) ("Not Enough Information"):

- Type: second order;
- Category: contradiction;
- Informal semantics: π_2 is an utterance that expresses the fact that its emitter does not have enough information to answer the question π_1;
- Example:
  π_1: *M*: What author are you interested in in this field?
  π_2: *U*: I don't know; what do you have?

11. *Alternation*(π_1, π_2):

- Type: third order;
- Category: confirmation;
- Informal semantics: this relation is equivalent to the logical "OR";
- Example:
  π_1: *M*: I can either lend you the book 'X',
  π_2: *M*: or show you the DVD 'Y', on this subject.

12. *Background*(π_1, π_2):

- Type: third order;
- Category: confirmation;
- Informal semantics: π_2 gives background information with respect to π_1;
- Example:
  π_1: *M*: I can lend you this book 'X'.
  π_2: *M*: You can make up to three reservation at one time.

13. *Consequence*(π_1, π_2):

- Type: third order;
- Category: confirmation;
- Informal semantics: this relation is equivalent to the logical implication;
- Example:
  π_1: *M*: If we give you the book 'X',
  π_2: *M*: then, the library will not have any more copy of it left.

14. *Elaboration*(π_1, π_2):

- Type: third order;
- Category: confirmation;
- Informal semantics: π_2 elaborates on π_1, so that they share the same topic;
- Example:
  π_1: *M*: I cannot lend you book 'X'.
  π_2: *M*: It is the only one copy in the library.

15. *Narration*(π_1, π_2):

- Type: third order;
- Category: confirmation;
- Informal semantics: π_2 temporally follows π_1, in the same discourse;
- Example:
  π_1: *M*: You reach the hall entitled "Scandinavian literature".
  π_2: *M*: Then, you go to shelf "Andersen".

16. *Contrast*(π_1, π_2):

- Type: third order;
- Category: contradiction;
- Informal semantics: π_1 and π_2 share the same topic, but present a contrastive element, one with respect to the other;
- Example:
  π_1: *M*: I can give you book 'X'.
  π_2: *M*: However, book 'Y' is available only for local access.

17. *Parallel*(π_1, π_2):

- Type: third order;
- Category: confirmation;
- Informal semantics: π_1 and π_2 have the same semantic structure and topic;
- Example:
  π_1: *M*: I can give you book 'X'.
  π_2: *M*: I can give book 'Y' as well.

In order to render this fragment of SDRT operational from a human-computer interaction perspective, the semantics of the 17 rhetorical relations chosen have been expressed in a first-order logic approximation of those provided in vanilla SDRT. This endeavour is motivated by the need to obtain a computationally tractable rhetorical structuring component for utterance generation in dialogue. The main idea resides in using a set of task-independent **discourse predicates** for expressing the semantics of the rhetorical relations. The formal aspects of the rhetorical structuring component are presented in [Popescu *et al.*, 2007a, Popescu *et al.*, 2007b]. In these papers, detailed examples illustrating the approach and the mechanism for updating the SDRS reflecting the dialogue between a user and a computer are provided. This is why a description of these aspects will not be given here.

As for the manner whereby the discourse structure is updated (i.e., the simple_update/2 procedure), this is described in thorough details in several papers: in [Popescu *et al.*, 2007a] we provide a baseline rhetorical structure updating algorithm, quadratic in the number of utterances already produced in dialogue; in [Popescu *et al.*, 2007b] we provide an optimized version of the baseline algorithm, whereby constraints induced by speech acts are used to yield rhetorical structures that are more in accord to SDRT specifications and human intuitions.

References

[Asher and Lascarides, 2003] Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*. Cambridge University Press.

[Caelen and Xuereb, 2007] Jean Caelen and Anne Xuereb. 2007. *Interaction et Pragmatique - jeux de dialogue et de langage*. Editions Hermès-Lavoisier, Paris.

[Cavazza *et al.*, 2002] Marc Cavazza, Fred Charles and Steven J. Mead. 2001. Planning Characters' Behaviour in Interactive Storytelling. *Proceedings of JVCA 2002*, 2002.

[Cristea *et al.*, 1998] Dan Cristea, Nancy Ide and Laurent Romary. 1998. Veins Theory. An Approach to Global Cohesion and Coherence. *Proceedings of COLING/ACL 1998*, Montreal, Canada, 1998. ACL, 1998.

[Ginzburg *et al.*, 2005] Johnathan Ginzburg and Raquel Fernandez. 2005. From Dialogue to Multilogue.... *Proceedings of ACL 2005*, 2005.

[Larson *et al.*, 2000] Stephen Larsson and David Traum. 2000. Information State and Dialogue Management in the TRINDI Dialogue Move Engine Toolkit. *Journal of Natural Language Engineering*, vol. 1, no. 1, 2000.

[McTear, 2002] Michael F. McTear. 2002. Spoken Language Technology: Enabling the Conversational User Interface. *ACM Computer Surveys*, 34(1).

[Nguyen and Caelen, 2005] Hoa Nguyen and Jean Caelen. 2005. Multi-session Management in Human-Computer Dialogue. *Proceedings of IBERAMIA 2004*, 2004.

[Popescu, 2007] Vladimir Popescu. 2007. Pronominal Anaphora Generation in Human-Computer Dialogue, Using a Semantic-Pragmatic Approach. *Proceedings of the Doctoral Consortium at the 8th FoLLI Eurolan Summer School*, "Al. I. Cuza" University Press, Iasi, Romania.

[Popescu *et al.*, 2007a] Vladimir Popescu, Jean Caelen and Corneliu Burileanu. 2007a. Logic-Based Rhetorical Structuring Component in Natural Language Generation for Human-Computer Dialogue. *Text, Speech and Dialog 2007*, LNAI, Springer, Heidelberg, 2007.

[Popescu *et al.*, 2007b] Vladimir Popescu, Jean Caelen and Corneliu Burileanu. 2007b. Using Speech Acts in Logic-Based Rhetorical Structuring for Natural Language Generation in Human-Computer Dialogue. *Proceedings of SIGDIal 2007*, Antwerp, Belgium, 2007.

[Popescu-Belis *et al.*, 2007] Andrei Popescu-Belis and Sandrine Zufferey. 2007. Contrasting the Automatic Identification of Two Discourse Markers in Multi-Party Dialogues. *Proceedings of SIGDIal 2007*, Antwerp, Belgium, 2007.

[Power, 1979] Richard Power. 1979. The Organisation of Purposeful Dialogues. *Linguistics*, 17:107-152, 1979.

[Schlangen *et al.*, 2001] D. Schlangen, A. Lascarides and A. Copestake. 2001. Resolving Underspecification Using Discourse Information. *Proceedings of Bi-Dialog 2001*, Springer.

[Thue *et al.*, 2007] David Thue, Vadim Bulitko, Marcia Spetch and Eric Wasylishen. 2007. Interactive Storytelling: A Player Modelling Approach. *Proceedings of AIIDE 2007*, 2007. AAAI, 2007.

[Vanderveken1990-1991] Daniel Vanderveken. 1990-1991. *Mening and Speech Acts*. Cambridge University Press.