



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Commitments to Preferences in Dialogue

Citation for published version:

Cadilhac, A, Asher, N, Benamara, F & Lascarides, A 2011, Commitments to Preferences in Dialogue. in *Proceedings of the SIGDIAL 2011 Conference*. Association for Computational Linguistics, Portland, Oregon, pp. 204-215. <<http://www.aclweb.org/anthology/W/W11/W11-2023>>

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Publisher's PDF, also known as Version of record

Published In:

Proceedings of the SIGDIAL 2011 Conference

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



Commitments to Preferences in Dialogue

Anais Cadilhac*, Nicholas Asher*, Farah Benamara*, Alex Lascarides**
*IRIT, University of Toulouse, **School of Informatics, University of Edinburgh

Abstract

We propose a method for modelling how dialogue moves influence and are influenced by the agents' preferences. We extract constraints on preferences and dependencies among them, even when they are expressed indirectly, by exploiting discourse structure. Our method relies on a study of 20 dialogues chosen at random from the Verbmobil corpus. We then test the algorithms predictions against the judgements of naive annotators on 3 random unseen dialogues. The average annotator-algorithm agreement and the average inter-annotator agreement show that our method is reliable.

1 Introduction

Dialogues are structured by various moves that the participants make—e.g., answering questions, asking follow-up questions, elaborating prior claims, and so on. Such moves come with commitments to certain attitudes such as intentions and preferences. While mapping utterances to their underlying *intentions* is well studied through the application of plan recognition techniques (e.g., Grosz and Sidner (1990), Allen and Litman (1987)), game-theoretic models of rationality generally suggest that intentions result from a deliberation to find the optimal tradeoff between one's *preferences* and one's *beliefs* about possible outcomes (Rasmusen, 2007). So mapping dialogue moves to *preferences* is an important task: for instance, they are vital in decisions on how to re-plan and repair should the agents' current plan fail, for they inform the agents about the relative importance of their various goals. Classical game theory, however, demands a complete and cardinal representation of preferences for the optimal intention to be defined. This is not realistic for modelling dialogue because agents often lack complete information about preferences prior to talking: they learn about the domain, each other's preferences and even their own preferences through dialogue exchange. For instance, utterance (1) implies that the speaker wants to go to the mall given that he wants to eat, but we do not

know his preferences over “go to the mall” if he does not want to eat.

- (1) I want to go to the mall to eat something.

Existing formal models of dialogue content either do not formalise a link between utterances and preferences (e.g., Ginzburg (to appear)), or they encode such links in a typed feature structure, where desire is represented as a feature that takes conjunctions of values as arguments (e.g., Poesio and Traum (1998)), making the language too restricted to express *dependencies* among preferences of the kind we just described. Existing implemented dialogue systems likewise typically represent goals as simple combinations of values on certain information ‘slots’ (e.g., He and Young (2005), Lemon and Pietquin (2007)); thus (1) yields a conjunction of preferences, to go to the mall and to eat something. But such a system could lead to suboptimal dialogue moves—e.g., to help the speaker go to the mall even if he has already received food.

What's required, then, is a method for extracting partial information about preferences and the dependencies among them that are expressed in dialogue, perhaps indirectly, and a method for exploiting that partial information to identify the next optimal action. This paper proposes a method for achieving these tasks by exploiting discourse structure.

We exploited the corpus of Baldridge and Lascarides (2005a), who annotated 100 randomly chosen spontaneous face-to-face dialogues from the Verbmobil corpus (Wahlster, 2000) with their discourse structure according to Segmented Discourse Representation Theory (SDRT, Asher and Lascarides (2003))—these structures represent the types of (relational) speech acts that the agents perform. Here's a typical fragment:

- (2) a. A: Shall we meet sometime in the next week?
b. A: What days are good for you?
c. B: Well, I have some free time on almost every day except Fridays.

- d. *B*: In fact, I'm busy on Thursday too.
- e. *A*: So perhaps Monday?

Across the corpus, more than 30% of the discourse units are either questions or assertions that help to elaborate a plan to achieve the preferences revealed by a prior part of the dialogue—these are marked respectively with the discourse relations *Q-Elab* and *Plan-Elab* in SDRT, and utterances (2b) and (2e) and the segments (2c) and (2d) invoke these relations (see Section 2). Moreover, 10% of the moves revise or correct prior preferences (like (2d)). We will model the interaction between dialogue content and preferences in two steps. The first maps utterances and their rhetorical connections into a partial description of the agents' preferences. The mapping is *compositional* and *monotonic* over the dialogue's logical form (i.e., the description of preferences for an extended segment is defined in terms of and always subsumes those for its subsegments): it exploits recursion over discourse structure. The descriptions partially describe *ceteris paribus preference nets* or CP-nets with Boolean variables (Boutillier et al., 2004). We chose CP-nets over alternative logics of preferences, because they provide a compact, computationally efficient, qualitative and relational representation of preferences and their dependencies, making them compatible with the kind of partial information about preferences that utterances reveal. Our mapping from the logical form of dialogue to partial descriptions of Boolean CP-nets proceeds in a purely linguistic or domain independent way (e.g., it ignores information such as Monday and Tuesday cannot co-refer) and will therefore apply to dialogue generally and not just *Verbmobil*.

In a second stage, we “compress” and refine our description making use of constraints proper to CP-nets (e.g., that preference is transitive) and constraints provided by the domain—in this case constraints about times and places, as well as constraints from deep semantics. This second step reduces the complexity of inferring which CP-net(s) satisfy the partial description and allows us to identify the minimal CP-net that satisfies the domain-dependent description of preferences. We can thus exploit dependencies between dialogue moves and mental states in a compact, efficient and intuitive way.

We start by motivating and describing the semantic representation of dialogue from which our CP-net descriptions and then our CP-nets will be constructed.

2 The Logical Form of Dialogue

Our starting point for representing dialogue content is SDRT. Like Hobbs et al. (1993) and Mann and Thompson (1987), it structures discourse into units that are linked together with *rhetorical relations* such as *Explanation*, *Question Answer Pair (QAP)*, *Q-Elab*, *Plan-Elab*, and so on. Logical forms

in SDRT consist of *Segmented Discourse Representation Structures (SDRSs)*. As defined in Asher and Lascarides (2003), an SDRS is a set of labels representing discourse units, and a mapping from each label to an SDRS-formula representing its content—these formulas are based on those for representing clauses or elementary discourse units (EDUS) plus rhetorical relation symbols between labels. Lascarides and Asher (2009) argue that to make accurate predictions about acceptance and denial, both of which can be implicated rather than linguistically explicit, the logical form of dialogue should track each agent's commitments to content, including rhetorical connections. They represent a dialogue turn (where turn boundaries occur whenever the speaker changes) as a set of SDRSs—one for each agent representing all his current commitments, from the beginning of the dialogue to the end of that turn. The representation of the dialogue overall—a Dialogue SDRS or DSDRS—is that of each of its turns. Each agent constructs the SDRSs for all other agents as well as his own. For instance, (2) is assigned the DSDRS in Table 1, with the content of the EDUS omitted for reasons of space (see Lascarides and Asher (2009) for details). We adopt a convention of indexing the root label of the n^{th} turn, spoken by agent d , as nd ; and $\pi : \phi$ means that ϕ describes π 's content (we'll sometimes also write ϕ_π to identify this description).

We now return to our example (2). Intuitively, (2a) commits *A* to a preference for meeting next week but it does so indirectly: the preference is not asserted, or equivalently entailed at the level of content from the semantics of *Q-Elab(a, b)*. Accordingly, responding with “I do too” (meaning “I want to meet next week too”) is correctly predicted to be highly anomalous. *A*'s SDRS for turn 1 in Table 1 commits him to the questions (2a) and (2b) because *Q-Elab* is veridical: i.e. *Q-Elab(a, b)* entails the dynamic conjunction $\phi_a \wedge \phi_b$. Since intuitively (2a) commits *A* to the implicature that he prefers next week, our algorithm for eliciting preferences from dialogue must ascribe this preference to *A* on the basis of his move *Q-Elab(a, b)*. Furthermore, *Q-Elab(a, b)* entails that any answer to (2b) must elaborate a plan to achieve the preference revealed by (2a); this makes ϕ_b paraphrasable as “What days next week are good for you?”, which does not add new preferences.

B's contribution in the second turn attaches to (2b) with *QAP* and also *Plan-Elab*—he answers with a non-empty extension for *what days*. Lascarides and Asher (2009) argue that this means that *B* is also committed to the illocutionary contribution of (2b), as shown in Table 1 by the addition of *Q-Elab(a, b)* to *B*'s SDRS. This addition commits *B* also to the preference of meeting next week, with his answer making the preference more precise: (2c) reveals that *B* prefers any day except Friday; by linking (2d) with *Plan-Correction* he retracts the preference for Thursday. This compels *A* to revise his inferences about

Turn	A's SDRS	B's SDRS
1	$\pi_{1A} : Q\text{-Elab}(a, b)$	\emptyset
2	$\pi_{1A} : Q\text{-Elab}(a, b)$	$\pi_{2B} : Q\text{-Elab}(a, b) \wedge QAP(b, \pi) \wedge Plan\text{-Elab}(b, \pi)$ $\pi : Plan\text{-Correction}(c, d)$
3	$\pi_{3A} : Q\text{-Elab}(a, b) \wedge QAP(b, \pi) \wedge Plan\text{-Elab}(b, \pi) \wedge$ $Plan\text{-Elab}(\pi, e)$	$\pi_{2B} : Q\text{-Elab}(a, b) \wedge QAP(b, \pi) \wedge Plan\text{-Elab}(b, \pi)$ $\pi : Plan\text{-Correction}(c, d)$

Table 1: The DSDRS for Dialogue (2).

B's preference for meeting on Thursday. A's *Plan-Elab* move (2e) in the third turn reveals another preference for Monday. This may not match his preferred day when the dialogue started: perhaps that was Friday. He may continue to prefer that day. But engaging in dialogue can compel agents to revise their commitments to preferences as they learn about the domain and each other.

The above discussion of (2) exhibits how different types of rhetorical *relations* between utterances rather than Searle-like speech acts like *question*, construed as a property of an utterance, are useful for encoding how preferences evolve in a dialogue and how they relate to one another. While the Grounding Acts dialogue model (Poesio and Traum, 1998) and the Question Under Discussion (QUD) model (Ginzburg, to appear) both have many attractive features, they do not encode as fine-grained a taxonomy of types of speech acts and their semantic effects as SDRT: in SDRT each rhetorical relation is a different kind of (relational) speech act, so that, for instance, the speech act of questioning is divided into the distinct types *Q-Elab*, *Plan-Correction*, and others. For the QUD model to encode such relations would require implicit questions of all sorts of different types to be included in the taxonomy, in which case the result may be equivalent to the SDRT taxonomy of dialogue moves. We have not explored this eventual equivalence here.

3 CP-nets and CP-net descriptions

A preference is standardly understood as an ordering by an agent over outcomes; at the very least it entails a comparison between one entity and another (outcomes being one sort of entity among others). As indicated in the introduction, we are interested in an ordinal definition of preferences, which consists in imposing an ordering over all (relevant) possible outcomes. Among these outcomes, some are acceptable for the agent, in the sense that the agent is ready to act in such a way as to realize them; and some outcomes are not acceptable. Amongst the acceptable outcomes, the agent will typically prefer some to others. Our method does not try to determine the most preferred outcome of an agent but follows rather the evolution of their commitments to certain preferences as the dialogue proceeds. To give an example, if an agent proposes to meet on a certain day X and at a certain time Y,

we infer that among the agent's acceptable outcomes is a meeting on X at Y, even if this is not his most preferred outcome (see earlier discussion of (2e)).

A CP-net (Boutilier et al., 2004) offers a compact representation of preferences. It is a graphical model that exploits conditional preferential independence so as to structure the decision maker's preferences under a *ceteris paribus* assumption.

Although CP-nets generally consider variables with a finite range of values, to define the mapping from dialogue turns to descriptions of CP-nets in a domain independent and compositional way, we use *Boolean* propositional variables: each variable describes an action that an agent can choose to perform, or not. We will then refine the CP-net description by using domain-specific information, transforming CP-nets with binary valued variables to CP-nets with multiple valued variables. This reduces the complexity of the evaluation of the CP-net by a large factor.

More formally, let V be a finite set of propositional variables and L_V the description language built from V via Boolean connectives and the constants \top (*true*) and \perp (*false*). Formulas of L_V are denoted by ϕ, ψ , etc. 2^V is the set of interpretations for V , and as usual for $M \in 2^V$ and $x \in V$, M gives the value *true* to x if $x \in M$ and *false* otherwise. Where $X \subseteq V$, let 2^X be the set of X -interpretations. X -interpretations are denoted by listing all variables of X , with a $\bar{}$ symbol when the variable is set to false: e.g., where $X = \{a, b, d\}$, the X -interpretation $M = \{a, d\}$ is expressed as $a\bar{b}d$.

A *preference relation* \succeq is a reflexive and transitive binary relation on 2^V with strict preference \succ defined in the usual way (i.e., $M \succeq M'$ but $M' \not\succeq M$). Note that preference orderings are not necessarily complete, since some candidates may not be comparable by a given agent. An agent is said to be indifferent between two options $M, M' \in 2^V$, written $M \sim M'$, if $M \succeq M'$ and $M' \succeq M$.

As we stated earlier, CP-nets exploit conditional preferential independence to compute a preferential ranking over outcomes:

Definition 1 *Let V be a set of propositional variables and $\{X, Y, Z\}$ a partition of V . X is conditionally preferentially independent of Y given Z if and only if $\forall z \in 2^Z, \forall x_1, x_2 \in 2^X$ and $\forall y_1, y_2 \in 2^Y$ we have: $x_1 y_1 z \succeq$*

x_2y_1z iff $x_1y_2z \succeq x_2y_2z$.

For each variable X , the agent specifies a set of *parent variables* $Pa(X)$ that can affect his preferences over the values of X . Formally, X is conditionally preferentially independent of $V \setminus (\{X\} \cup Pa(X))$. This is then used to create the CP-net.

Definition 2 Let V be a set of propositional variables. $\mathcal{X} = \langle \mathcal{G}, \mathcal{T} \rangle$ is a CP-net on V , where \mathcal{G} is a directed graph over V , and \mathcal{T} is a set of Conditional Preference Tables (CPTs) with indifference. That is, $T = \{\text{CPT}(X_j) : X_j \in V\}$, where $\text{CPT}(X_j)$ specifies for each instantiation $p \in 2^{Pa(X_j)}$ either $x_j \succ_p \bar{x}_j$, $\bar{x}_j \succ_p x_j$ or $x_j \sim_p \bar{x}_j$.

The following simple example illustrates these definitions. Suppose our agent prefers to go from Paris to Hong Kong by day rather than overnight. If he takes an overnight trip, he prefers a non stop flight, but if he goes by day, he prefers a flight with a stop. Figure 1 shows the associated CP-net. The variable T stands for the preference over the period of travel. Its values are T_d for a day trip and T_n for a night one. The variable St stands for the preference over stops. Its values are S for a trip with stops and \bar{S} without.

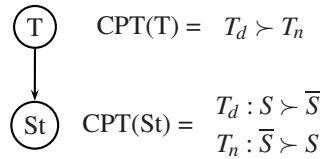


Figure 1: Travel CP-net

With CP-nets defined, we proceed to a description language for them. The description language formula $w \succ y(CPT)$ describes a CP-net where a CPT contains an entry of the form $w \succ_p y$ for some possibly empty list of parent variables p . A CP-net description is a set of such formulas. The CP-net $\mathcal{X} \models_{x_1, \dots, x_n} w \succ y(CPT)$ iff the CP-net \mathcal{X} 's CPT \mathcal{T} contains an entry $w \succ_{\bar{u}} y$ —also written $\bar{u} : w \succ y$ —where x_1, \dots, x_n figure in \bar{u} . Satisfaction of a description formula by a CP-net yields a notion of logical consequence between a CP-net description $\mathcal{D}\mathcal{X}$ and a description formula in the obvious way. Dialogue turns also sometimes inform us that certain variables enter into preference statements. We'll express the fact that the variables x_1, \dots, x_n are associated with discourse constituent π by the formula $x_1, \dots, x_n(P(\pi))$, where $P(\pi)$ refers to the partial description of the preferences expressed by the discourse unit π (see Section 4).

The description language allows us to impose constraints on the CP-nets that agents commit to without specifying the CP-net completely, as is required for utterances like (1). In section 6, we describe how to construct a minimal CP-net from a satisfiable CP-net description. One

can then use the *forward sweep* procedure for outcome optimisation (Boutilier et al., 2004). This is a procedure of *linear* complexity, which consists in instantiating variables following an order compatible with the graph, choosing for each variable (one of) its preferred values given the value of the parents.

4 From EDUs to Preferences

EDUs are described in SDRT using essentially Boolean formulas over labels (Asher and Lascarides, 2003); thus $\phi(\pi) \wedge \psi(\pi)$ means that ϕ and ψ describe aspects of π 's content. $\text{Not}(\pi_1, \pi) \wedge \phi(\pi_1)$ means that the logical form of the EDU π is of the form $\neg\pi_1$ and that π_1 is described by ϕ ; so π has the content $\neg\phi$. Our task is to map such descriptions of content into descriptions of preferences. Our preference descriptions will use Boolean connectives and operators over preference entries (e.g., of the form $x \succ y$): namely, $\&$, ∇ , \mapsto , and a modal operator \diamond . The rules below explain the semantics of preference operators (they are in effect defined in terms of the semantics of deontic attitudes and Boolean connectives) and how to recursively calculate preference descriptions from the EDU's logical structure.

Simple EDUs can provide *atomic* preference statements (e.g., *I want X* or *We need X*). This means that with this EDU the speaker commits to a preference for X . X will typically involve a Boolean variable and a preference entry for its CPT. $P(\pi)$ is the label of the preference description associated with discourse unit π . Hence for a simple EDU π , we have $X(P(\pi))$ as its description. Simple EDUs also sometimes express preferences in an indirect way (see (2a)).

More generally, P recursively exploits the logical structure of an EDU's logical form to produce an EDU *preference representation* (EDUPR). For instance, since the logical form of the EDU *I want fish and wine* features conjunction, likewise so does its preference description: $\phi \& \psi(P(\pi))$ means that among the preferences included in π , the agent prefers to have both ϕ and ψ and prefers either one if he can't have both.¹ We also have disjunctions (*let's meet Thursday or Friday*), and negations (*I don't want to meet on Friday*), whose preferences we'll express respectively as $\text{Thurs} \nabla \text{Fri}(P(\pi))$ and $\neg \text{Fri}(P(\pi))$.

Some EDUs express commitments to dependencies among preferences. For example, in the sentence *What about Monday, in the afternoon?*, there are two preferences: one for the day Monday, and, given the Monday preference, one for the time afternoon (of Monday), at least on one syntactic disambiguation. We represent this dependency as $\text{Mon} \mapsto \text{Aft}(P(\pi))$. Note that \mapsto is not expressible with just Boolean operators. Finally, EDUs can express commitment to preferences via free choice

¹The full set of rules also includes a stronger conjunction $\phi \Delta \psi(P(\pi))$ (the agent prefers both ϕ and ψ , but is indifferent if he can't have both).

modalities; *I am free on Thursday*, or $\diamond\text{Thurs}(P(\pi))$, tells us that Thursday is a possible day to meet. $\diamond\phi$ says that ϕ is an acceptable outcome (as described earlier, this means the agent is ready to act so as to realize an outcome that entails ϕ). Thus, $\diamond\phi(\pi)$ entails $\phi(\pi)$, and \diamond -embedded preferences obey reduction axioms permitting \diamond to be eliminated when combined with other preference operators. But a \diamond preference statement does affect a preference description when is is conjoined in Boolean fashion with another \diamond preference statement in an EDU or combined via a discourse relation like *Continuation*. This is because \diamond is a free choice modality and obeys the equivalence (3) below, which in turn yields a *disjunctive* preference $\phi \nabla \psi(P(\pi))$ from what appeared to be a conjunction.²

$$(3) \quad (\diamond\phi(P(\pi)) \wedge \diamond\psi(P(\pi))) \leftrightarrow \diamond(\phi \nabla \psi)(P(\pi))$$

The variables introduced by a discourse segment π are integrated into the CP-net description $\mathcal{D}\mathcal{N}$ via the operation $\text{Commit}(\pi, \mathcal{D}\mathcal{N})$. The following seven rules cover the different possible logical structures for the EDU preference representation. In the following, X, Y, Z, W denote propositional variables and ϕ, ψ propositional formulas from EDUPR. $\text{Var}(\phi)$ are the variables in ϕ , and \succ_X the preference relation describing $CPT(X)$. $\text{Sat}(\phi)$ (or $\text{non-Sat}(\phi)$) is a conjunction of literals from $\text{Var}(\phi)$ that satisfy (or do not satisfy) ϕ . $\text{Sat}(\phi) - X$ is the formula that results from removing the conjunct with X from $\text{Sat}(\phi)$.

1. Where $X(P(\pi))$ (X is a variable of $P(\pi)$, e.g., *I want X*), $\text{Commit}(\pi, \mathcal{D}\mathcal{N})$ adds the description $\mathcal{D}\mathcal{N} \models X \succ \bar{X}(CPT(X))$.³
2. Where $\phi \& \psi(P(\pi))$, $\text{Commit}(\pi, \mathcal{D}\mathcal{N})$ adds descriptions as follows:
 - For each $X \in \text{Var}(\phi)$, add $\text{Var}(\psi)$ to $\text{Pa}(X)$ and modify $CPT(X)$ as follows:
If $\text{Sat}_i(\psi), \text{Sat}_j(\phi) \vdash X$ (resp. \bar{X}), then $\text{Sat}_i(\psi), \text{Sat}_j(\phi) - X : X \succ \bar{X}$ (resp. $\bar{X} \succ X$), for all satisfiers i and j .
 - Similarly for each $Y \in \text{Var}(\psi)$.

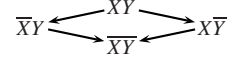
If ϕ and ψ are literals X and Y we get: $\mathcal{D}\mathcal{N} \models Y \succ \bar{Y}(CPT(Y))$ and $\mathcal{D}\mathcal{N} \models X \succ \bar{X}(CPT(X))$. Graphically, this yields the following preference relation (where one way arrows denote preference, two way

²We provide here the reduction axioms over preference descriptions

1. $\diamond(\phi \& \psi)(P(\pi)) \leftrightarrow (\phi \& \psi)(P(\pi))$
2. $\diamond(\phi \mapsto \psi)(P(\pi)) \leftrightarrow (\phi \mapsto \psi)(P(\pi))$
3. $\diamond(\phi \nabla \psi) \leftrightarrow (\phi \nabla \psi)(P(\pi))$
4. $\diamond\diamond\phi(P(\pi)) \leftrightarrow \diamond\phi(P(\pi))$

³Given our description language semantics, this means that any CP-net which satisfies the description $\mathcal{D}\mathcal{N}$ contains a preference table $CPT(X)$ with an entry $X \succ \bar{X}$ with at least one instantiation of the variables in $\text{Pa}(X)$.

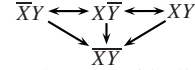
arrows denote indifference or equal preference, and no arrow means the options are incomparable):



3. Where $\phi \nabla \psi(P(\pi))$ (the agent prefers to have at least one of ϕ and ψ satisfied). If ϕ and ψ are X and Y , we get:

- $\text{Var}(X) \in \text{Pa}(\text{Var}(Y))$ and $\mathcal{D}\mathcal{N} \models X : Y \sim \bar{Y}(CPT(Y))$, $\mathcal{D}\mathcal{N} \models \bar{X} : Y \succ \bar{Y}(CPT(Y))$.
- $\text{Var}(Y) \in \text{Pa}(\text{Var}(X))$ and $\mathcal{D}\mathcal{N} \models Y : X \sim \bar{X}(CPT(X))$, $\mathcal{D}\mathcal{N} \models \bar{Y} : X \succ \bar{X}(CPT(X))$.

This corresponds to the following preference relation:



As before, the use of indifference allows us to find the best outcomes (XY , $X\bar{Y}$ and $\bar{X}Y$) easily.

4. Where $\phi \mapsto \psi(P(\pi))$ (the agent prefers that ϕ is satisfied and if so that ψ is also satisfied. If ϕ is not satisfied, it is not possible to define preferences on ψ). If ϕ and ψ are X and Y , we get:

- $\mathcal{D}\mathcal{N} \models X \succ \bar{X}(CPT(X))$
- $\text{Var}(X) \in \text{Pa}(\text{Var}(Y))$ and $\mathcal{D}\mathcal{N} \models X : Y \succ \bar{Y}(CPT(Y))$.

Note that this description is also produced by $\text{Elab}(\pi_i, \pi_j)$ below where $X(P(\pi_i))$ and $Y(P(\pi_j))$ (see rule 8). Thus the implication symbol \mapsto is a "shortcut" in that it represents elaborations whose arguments are in the same EDU.

5. Where $\diamond\phi(P(\pi))$ (the agent prefers a free choice of ϕ). Given the behaviour of \diamond , this reduces to treating $\phi(P(\pi))$.
6. Where $\neg\phi(P(\pi))$. We can apply rules 1-5 by converting $\neg\phi$ into conjunctive normal form.
7. Where $\phi(P(\pi)) \wedge \psi(P(\pi))$, with ϕ and ψ nonmodal, we simply apply the rule for ϕ and that for ψ .

5 From Discourse Structure to Preferences

We must now define how the agents' preferences, represented as a partial description of a CP-net, are built compositionally from the discourse structure over EDUs. The constraints are different for different discourse relations, reflecting the fact that the semantics of connections between segments influences how their preferences relate to one another.

We will add rules for defining Commit over labels π whose content ϕ_π express rhetorical relations $R(\pi_i, \pi_j)$ —indeed, we overload the notation and write $\text{Commit}(R(\pi_i, \pi_j), \mathcal{D}\mathcal{N})$. Since Commit applies compositionally, starting with the EDUs and working up

the discourse structure towards the unique root label of the SDRS, we can assume in our definition of $Commit(R(\pi_i, \pi_j), \mathcal{D}\mathcal{N})$ that the EDUPRs are already defined. We give rules for all the relations in the Verbmobil corpus, though we will be very brief with those that are less prevalent. A complete example using our rules is in appendix A.

IEExplanation, Elab, Plan-Elab, Q-elab

IEExplanation(π_i, π_j): i.e., π_j 's preferences explain π_i 's (e.g., see (1), where $P(\pi_i)$ would be going to the mall and $P(\pi_j)$ is eating something). With *Elab*(π_i, π_j) a preference in π_i is elaborated on or developed in π_j , as in: *I want wine. I want white wine.* That is, a preference for white wine depends on a preference for wine. *Plan-Elab*(π_i, π_j) means that π_j describes a plan for achieving the preferences expressed by π_i , and with *Q-Elab* we have a similar dependence between preferences, but the second constituent is a question (so often in practice this means preference commitments from π_i transfer from one agent to another).

Plan-Elab(π_j, π_i), *Elab*(π_j, π_i) and *IEExplanation*(π_i, π_j) all follow the same two-step rule, and so from the point of view of preference updates they are equivalent:

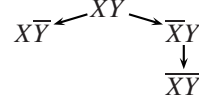
8. i Firstly, preference description $\mathcal{D}\mathcal{N}$ is updated according to $P(\pi_j)$ by applying $Commit(\pi_j, \mathcal{D}\mathcal{N})$, if π_j expresses a new preference. If not go to step (ii).
- ii. Secondly, description $\mathcal{D}\mathcal{N}$ is modified so that each variable in $P(\pi_i)$ depends on each variable in $P(\pi_j)$: i.e., $\forall X \in Var(P(\pi_i)), \forall Y \in Var(P(\pi_j)), Y \in Pa(X)$. Then, $\mathcal{D}\mathcal{N}$ is enriched according to $P(\pi_i)$, if π_i expresses a preference. If it does not, then end.

We now give some details concerning step (ii) above. To this end, let ϕ denote a formula with SDRS description predicates, ϕ' its corresponding boolean (preference) formula and $\bar{\phi}'$ its negation. Then for $\phi = Y$, we define $\phi' = Y$ and $\bar{\phi}' = \bar{Y}$; for $\phi = Y \mapsto Z$ we define $\phi' = Y \wedge Z$ and $\bar{\phi}' = \bar{Y} \vee \bar{Z}$; and for $\phi = Y \nabla Z$ and $\phi = Y \& Z$, we have $\phi' = Y \vee Z$ and $\bar{\phi}' = \bar{Y} \wedge \bar{Z}$.

- a. $X(P(\pi_i))$ and $\phi(P(\pi_j))$. The agent explains his preferences on X by ϕ . So, if no preferences on X are already defined, ϕ is a reason to prefer X . That is, $\mathcal{D}\mathcal{N} \models \phi' : X \succ \bar{X}(CPT(X))$. However, it is not possible to define preferences on X if ϕ is false. If, on the other hand, preferences on X are already defined, the agent prefers X if ϕ is satisfied, and does not modify his preferences otherwise—i.e., $\succ_{X, \phi'} = X \succ \bar{X}, \succ_{X, \bar{\phi}'} = \succ_X$.⁴

⁴If we have \succ_X such that $Z: \bar{X} \succ X, \bar{Z}: X \succ \bar{X}, \succ_{X, \phi'}$ represents preferences defined by $Z \wedge \phi'$ and $\bar{Z} \wedge \bar{\phi}'$, whereas $\succ_{X, \bar{\phi}'}$ represents preferences defined by $Z \wedge \bar{\phi}'$ and $\bar{Z} \wedge \phi'$.

For $\phi = Y$, if \succ_X is not already defined, we obtain the following preference relation (no information on the preference for X if Y is false makes $X\bar{Y}$ and $\bar{X}Y$ incomparable):

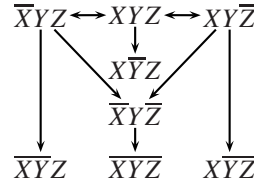


- b. $X \nabla Z(P(\pi_i))$ and $\phi(P(\pi_j))$. The agent explains his preferences $X \nabla Z$ by ϕ : he wants to satisfy X or Z if ϕ is satisfied.

First, we set $Var(Z) \in Pa(Var(X)), Var(X) \in Pa(Var(Z))$. If \succ_X is not already defined, we have: $\mathcal{D}\mathcal{N} \models \phi' \wedge Z: X \sim \bar{X}(CPT(X)), \mathcal{D}\mathcal{N} \models \phi' \wedge \bar{Z}: X \succ \bar{X}(CPT(X))$.

Otherwise, $\succ_{X, \phi' \wedge Z} = X \sim \bar{X}, \succ_{X, \phi' \wedge \bar{Z}} = X \succ \bar{X}, \succ_{X, \bar{\phi}' \wedge Z} = \succ_{X, \bar{\phi}' \wedge \bar{Z}} = \succ_X$.

$CPT(Z)$ is defined as $CPT(X)$ by inverting X and Z . For $\phi = Y$, if \succ_X and \succ_Z are not already defined, we obtain the following preference relation (again, the lack of preference information on X and Z when Y is false yields incomparability among states where Y is false):



- c. $X \& Z(P(\pi_i))$ and $\phi(P(\pi_j))$. The agent explains his preferences on $X \& Z$ by ϕ .

- If \succ_X is not already defined, we have: $\mathcal{D}\mathcal{N} \models \phi' : X \succ \bar{X}(CPT(X))$.
Otherwise, $\succ_{X, \phi'} = X \succ \bar{X}, \succ_{X, \bar{\phi}'} = \succ_X$,
- $CPT(Z)$ is defined as $CPT(X)$ by replacing X by Z .

- d. $X \mapsto Z(P(\pi_i))$ and $\phi(P(\pi_j))$. The agent explains his preferences on $X \mapsto Z$ by ϕ : he wants to satisfy X and after Z if ϕ is satisfied.

If \succ_X is not already defined, we have $\mathcal{D}\mathcal{N} \models \phi' : X \succ \bar{X}(CPT(X))$ and we set $Var(X) \in Pa(Var(Z))$.⁵ If \succ_Z is not yet defined, we have: $\mathcal{D}\mathcal{N} \models \phi' \wedge X : Z \succ \bar{Z}(CPT(Z)), \mathcal{D}\mathcal{N} \models \phi' \wedge \bar{X} : Z \sim \bar{Z}(CPT(Z))$. Else, $\succ_{Z, (\phi' \wedge X)} = Z \succ \bar{Z}, \succ_{Z, (\phi' \wedge \bar{X})} = Z \sim \bar{Z}, \succ_{Z, (\bar{\phi}' \wedge X)} = \succ_{Z, (\bar{\phi}' \wedge \bar{X})} = \succ_Z$.

- e. $\psi(P(\pi_i))$ and $\phi(P(\pi_j))$. We can apply rules 8 by decomposing ψ .

⁵Otherwise, there is no need to modify \succ_X . This is what we call a 'partial elaboration'. Variables that were evoked since preferences on X were introduced are parents of Z but not of X . For example, if an agent commits to a preference for *Monday* then *Afternoon*, and later in the discourse he commits to *2oclock*, then *Afternoon* is *2oclock*'s parent but not *Monday*'s.

- f. $\diamond(\psi)(P(\pi_i))$ and $\diamond(\phi)(P(\pi_j))$. We treat this like a free choice EDU (see rule 5).
- g. $\diamond(\psi)(P(\pi_i))$ and $\phi(P(\pi_j))$, where ϕ is non modal. We treat this like $\psi(P(\pi_i))$ and $\phi(P(\pi_j))$ (see rule 8.e)

Let's briefly look at how the rule changes for $Q\text{-elab}_A(\pi_1, \pi_2)$ (where the subscript A identifies the speaker of π_2):

9. $Q\text{-Elab}_A(\pi_1, \pi_2)$ implies that we update A 's CP-net description \mathcal{DN} by applying the rule for $Elab(\pi_1, \pi_2)$, where if π_2 expresses no preferences on their own, we simply make the $P(\pi_2)$ description equal to the $P(\pi_1)$ description. Thus A 's CP-net description is updated with the preferences expressed by utterance π_1 , regardless of who said π_1 .

QAP Answers to questions affect preferences in complex ways:

10. The first case concerns yes/no questions and there are two cases, depending on whether B replies *yes* or *no*:
- Yes** $QAP_B(\pi_1, \pi_2)$ where π_2 is *yes*. B 's preference descriptions are updated by applying $Commit(Elab_B(\pi_1, \pi_2), \mathcal{DN})$ (and so B 's preference description include preferences expressed by π_1 and π_2).
- No** $QAP_B(\pi_1, \pi_2)$ where π_2 is *no*. If $P(\pi_1)$ and $P(\pi_2)$ are consistent, then B 's preference descriptions are updated by applying $Commit_B(Elab_B(\pi_1, \pi_2), \mathcal{DN})$; otherwise, they are updated by applying $Commit(Correction(\pi_1, \pi_2), \mathcal{DN})$ (see rule 13).
11. When π_1 is a *wh*-question and $QAP_B(\pi_1, \pi_2)$, B 's preferences over variables in π_1 and π_2 are exactly the same as the ones defined for a yes/no question where the answer is *yes*. Variables in π_2 will refine preferences over variables in π_1 . So, B 's preference descriptions are updated by applying $Commit_B(Elab_B(\pi_1, \pi_2), \mathcal{DN})$.

In previous rules, it is relatively clear how to update the preference commitments. However, in some cases it's not clear what the answer in a QAP targets: in *Could we meet the 25 in the morning? No, I can't.*, we do not know if *No* is about the 25 and the morning, or only about the morning. So, we define the following rule for managing cases where the *target* is unknown :

12. If we know the target, we can change the description of the CP-net. Otherwise, we wait to learn more.

Correction and Plan-Correction allow a speaker to rectify a prior commitment to preferences. Self-corrections also occur in the corpus: *I could do it on the 27th. No I can not make it on the 27th, sorry I have a seminar.* *Correction* and *Plan-Correction* can have several effects on the preferences. For instance, they can correct preference entries. That is, given $Correction(\pi_1, \pi_2)$, some variables in $P(\pi_1)$ are replaced by variables in $P(\pi_2)$ (in the self-correction example, every occurrence of 27 in $P(\pi_1)$ is replaced with $\overline{27}$ and vice versa). We have a set of rules of the form $X \leftarrow \{Y_1, \dots, Y_m\}$, which means that the variable $X \in Var(P(\pi_1))$ is replaced by the set of variables $\{Y_1, \dots, Y_m\} \subseteq Var(P(\pi_2))$. We assume that X can't depend on $\{Y_1, \dots, Y_m\}$ before the *Correction* is performed. Then replacement proceeds as follows:

13. If $Pa(X) = \emptyset$, we add the description $\mathcal{DN} \models Y_k \succ \overline{Y}_k(CPT(Y_k))$ for all $k \in \{1, \dots, m\}$ and remove $X \succ \overline{X}(CPT(X))$ (or $\overline{X} \succ X(CPT(X))$). Otherwise, we replace every description of $CPT(X)$ with an equivalent statement using Y_k (to describe $CPT(Y_k)$), for all $k \in \{1, \dots, m\}$.

The specific target of the correction behaves similarly to the target of a QAP . In some cases we don't know the target, in which case we apply rule 12.

Plan-Correction can also lead to the modification of an agent's own plan because of other agent's proposals. In this case it corrects the list of parent variables on which a preference depends. We call that list of variables the *operative variables*. Once the operative variables are changed, *Plan-Correction* can elaborate a plan if some new preferences are expressed. For example, all agents have agreed to meet next week, so in their CP-net description, there is the entry $Week1 \succ \overline{Week1}$. Then discussion shows that their availabilities are not compatible and one of them says "okay, that week is not going to work.". That does not mean the agent prefers $\overline{Week1}$ to $Week1$ because both agreed on $Week1$ as preferable. Rather, $Week1$ has been removed as an operative variable in the following discourse segments. This leads us to the following rule:

14. For $Plan-Correction(\pi_1, \pi_2)$ which corrects the list of parent variables, the operative variable list becomes the intersection of all $Pa(X)$ where $X \in Var(P(\pi_1))$. We can now apply $Commit(Plan-Elab(\pi_1, \pi_2), \mathcal{DN})$, if $P(\pi_2)$ contains some new preferences ϕ . If the CPT affected by a rule has no entry for the current operative variable list O , then $O : \phi$ has to be added to \mathcal{DN} .

Continuation, Contrast and **Q-Cont** pattern with the rule for *Elab*. **Alternation** patterns with rule 8.b.⁶ **Explanation, Explanation***, **Result, Qclar** (clarification question), **Commentary, Summary** and **Acknowledgment**

⁶The rule for Alternative questions like *Do you want fish or chicken?* is a special case yielding $\phi \nabla \psi(P(\pi))$, but we don't offer details here.

either do nothing or have the same effect on preference elicitation as *Elab*. Sometimes, adding these preferences via the *Elab* rule may yield an unsatisfiable CP-net description, because an implicit correction is involved. If an evaluation of the CP-net (see next section) is performed after a processing of one of these rules shows that the CP-net description is not satisfiable, then we apply the rule 13, associated with *Correction*.

6 From Descriptions to Models

Each dialogue turn adds constraints monotonically to the descriptions of the CP-nets to which the dialogue participants commit. We have interpreted each new declared variable in our rules as independent, which allows us to give a domain independent description of preference elicitation. However, when it comes to evaluating a CP-net description for satisfiability, we need to take into account various axioms about preference (irreflexivity and transitivity), and axioms for the domain of conversation: in our case, temporal designations (Wednesdays are not Tuesdays and so on). This typically adds dependencies among the variables in the description. In the case of the Verbmobil domain, since the variable *Monday* means essentially "to meet on Monday", *Monday* implies *Meet*, and this must be reflected via a dependency in the CP-net: we must view the variable *Meet* as filling a hidden slot in the variable *Monday* in the preference description, $Meet : Mon \succ \overline{Mon}$. This likewise allows us to fill in the negative clauses of the CP-net description: we can now infer that $\overline{Meet} : \overline{Mon} \succ Mon$. These axioms also predict certain preference descriptions to be unsatisfiable. For instance, if we have $Mon \succ \overline{Mon}$, our axioms imply $Mon \succ Tues$, $Mon \succ Wed$, etc. At this point we can calculate, *ceteris paribus*, inconsistencies on afternoons and mornings of particular days.

Domain knowledge also allows us to collapse Boolean valued variables that all denote, say, days or times of the day into multiple valued variables. So for instance, our domain independent algorithm from dialogue moves to preference descriptions might yield:

$$(4) \quad Meet \wedge \overline{31.01} \wedge \overline{30.01} \wedge 02.02 : am \succ \overline{am}$$

Domain knowledge collapses all Boolean variables for distinct days into one variable with values for days to get:

$$(5) \quad Meet \wedge 02.02 : am \succ pm$$

This leads to a sizeable reduction in the set of variables that are used in the CP-net.

We can test any CP-net description for satisfiability by turning the description formulas into CP-net entries. Our description automatically produces a directed graph over the parent variables. We have to check that the \succ statements form an irreflexive and transitive relation and that each variable introduced into the CP-net has a preference

entry consistent given these constraints. If the description does not yield a preference entry for a given variable X , we will add the indifference formula $X \sim \overline{X}$ as the entry. If our CP-net description meets these requirements, this procedure yields a minimal CP-net. Testing for satisfiability is useful in eliciting preferences from several discourse moves like *Explanation*, *Qclar* or *Result*, since in the case of unsatisfiability, we will exploit the Correction rule 13 with these moves.

7 Evaluation of the proposed method

We evaluate our method by testing it against the judgements of three annotators on three randomly chosen unseen test dialogues from the Verbmobil corpus. The test corpus contains 75 EDUs and the proportion of discourse relations is the same as in the corpus overall. The three annotators were *naive* in the sense that they were not familiar with preference representations and preference reasoning strategies. For each dialogue segment, we checked if the judges had the same intuitions that we did on: (i) how commitments to preferences are extracted from EDUs, and (ii) how preferences evolve through dialogue exchange.

The judges were given a manual with all the instructions and definitions needed to make the annotations. For example, the manual defined preference to be "a notion of comparison between one thing at least one other". The manual also instructs annotators to label each EDU with the following four bits of information: (1) preferences (if any) expressed in the EDU; (2) dependencies between preferences expressed in the EDU; (3) dependencies between preferences in the current EDU and previous ones; and (4) preference evolution (namely, the appearance of a new factor that affects preferred outcomes, update to preferences over values for an existing factor, and so on). For each of these four components, example dialogues were given for each type of decision they would need to make, and instructions were given on the format in which to code their judgements. Appendix A shows an example of an annotated dialogue.

Table 2 presents results of the evaluation of (i). For each EDU, we asked the annotator to list the preferences expressed in the EDU and we compared the preferences extracted by each judge with those extracted by our algorithm. The triple (a, b, c) respectively indicates the proportion of common preferences (two preference sets Γ_i and Γ_j are common if $(\Gamma_i = \Gamma_j)$ or $(\exists x \in \Gamma_i, y \in \Gamma_j, x \rightarrow y)$)—for example, the preference $MeetBefore2 \succ MeetAt2$ implies $\overline{MeetAt2} \succ \overline{MeetAt2}$, the proportion of preferences that one judge extracts and the other judge or our algorithm misses and the proportion of preferences missed by one judge and extracted by the other judge or by our algorithm. The average annotator-algorithm agreement (AAA) is 75.6% and the average inter-annotator agree-

	Our algorithm	J1	J2	J3	% of EDUs that commit to preferences
Our algorithm		(83, 4, 13)	(91, 0, 9)	(91, 0, 9)	76%
J1	(83, 13, 4)		(85, 7, 8)	(91, 4, 5)	80%
J2	(91, 9, 0)	(85, 8, 7)		(92, 4, 4)	86%
J3	(91, 9, 0)	(91, 5, 4)	(92, 4, 4)		84%

Table 2: Evaluating how preferences are extracted from EDUs.

	Our algorithm	J1	J2	J3
Our algorithm		(85, 71)	(96, 100)	(93, 86)
J1	(85, 71)		(89, 71)	(91, 86)
J2	(96, 100)	(89, 71)		(98, 86)
J3	(93, 86)	(91, 86)	(98, 86)	

Table 3: Evaluating how preferences evolve through dialogue.

ment (IAA) is 77.9%; this shows that our method for extracting preferences from EDUs is reliable.

The evaluation (ii) proceeds as follows. For each EDU, we ask the judge if the segment introduces new preferences or if it updates, corrects or deletes preferences committed in previous turns. As in (i), judges have to justify their choices. Table 3 presents the preliminary results where the couple (a,b) indicates respectively the proportion of common elaborations (preference updates or new preferences) and the proportion of common corrections. Since elaboration is also applied in case of other discourse relations (e.g., *Q-Elab*), the measure a evaluates the rules 8, 9, 10 (**yes**) and 11. Similarly, the measure b evaluates the rules 10 (**no**), 13 and 14. We obtain AAA=91% IAA=92.7% for elaboration and AAA=85.7% IAA=81% for correction.

8 Conclusion

We have proposed a compositional method for eliciting preferences from dialogue consisting of a domain-independent algorithm for constructing a partial CP-net description of preferences, followed by a domain-specific method for identifying the minimal CP-net satisfying the partial description and domain constraints. The method supports qualitative and partial information about preferences, with CP-nets benefiting from linear algorithms for computing the optimal outcome from a set of preferences and their dependencies. The need to compute intentions from partially defined preferences is crucial in dialogue, since preferences are acquired and change through dialogue exchange.

Our work partially confirms that CP-nets have a certain naturalness, as the map from dialogue moves to preferences using the CP-net formalism is relatively intuitive. The next step is to implement our method. This depends

on extracting discourse structure from text, which, though difficult, is becoming increasingly tractable for simple domains (Baldrige and Lascarides, 2005b). We plan to extract CP-net descriptions from EDUs and to evaluate these descriptions using "multi-valued variables" automatically. We will then evaluate our method on a large number of dialogues.

Our work here is also and more generally a first step towards modelling the complex interaction between what agents say, what their preferences are, and what they take the preferences of other dialogue agents to be. It leads to a conception of dialogue that's more general than one based purely on Gricean cooperative principles (Grice, 1975). On a purely Gricean approach, conversation is cooperative in at least two ways: a basic level concerning the conventions that govern linguistic meaning (basic cooperativity); and a level concerning shared attitudes towards what is said, including shared intentions (content cooperativity). While basic cooperation is needed for communication to work at all, content cooperativity involves strongly cooperative axioms like Cooperativity (interlocutors normally adopt the speaker's intentions) (Allen and Litman, 1987, Grosz and Sidner, 1990, Lochbaum, 1998). Our approach allows for divergent preferences and divergent intentions, i.e. conversations that aren't based on content cooperativity. This will allow us to exploit information about conflicting agents' preferences and game-theoretic techniques that are inherent in the logics of CP-nets for computing optimal moves (Bonzon, 2007). And in contrast to Franke et al. (2009), who analyse conversations where content cooperativity doesn't hold using a game-theoretic framework, our approach allows for partial and qualitative representations of preferences rather than demanding complete and quantitative representations of them.

References

- J. Allen and D. Litman. A plan recognition model for subdialogues in conversations. *Cognitive Science*, 11(2):163–200, 1987.
- N. Asher and A. Lascarides. *Logics of Conversation*. Cambridge University Press, 2003.
- J. Baldridge and A. Lascarides. Annotating discourse structures for robust semantic interpretation. In *Proceedings of the Sixth International Workshop on Computational Semantics (IWCS)*, Tilburg, The Netherlands, 2005a.
- J. Baldridge and A. Lascarides. Probabilistic head-driven parsing for discourse structure. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL)*, pages 96–103, 2005b.
- E. Bonzon. *Modélisation des Interactions entre Agents Rationnels: les Jeux Booléens*. PhD thesis, Université Paul Sabatier, Toulouse, 2007.
- C. Boutilier, R.I. Brafman, C. Domshlak, H.H. Hoos, and David Poole. Cp-nets: A tool for representing and reasoning with conditional *ceteris paribus* preference statements. *Journal of Artificial Intelligence Research*, 21:135–191, 2004.
- M. Franke, T. de Jager, and R. van Rooij. Relevance in cooperation and conflict. *Journal of Logic and Language*, 2009.
- J. Ginzburg. *The Interactive Stance: Meaning for Conversation*. CSLI Publications, to appear.
- H. P. Grice. Logic and conversation. In P. Cole and J. L. Morgan, editors, *Syntax and Semantics Volume 3: Speech Acts*, pages 41–58. Academic Press, 1975.
- B. Grosz and C. Sidner. Plans for discourse. In J. Morgan P. R. Cohen and M. Pollack, editors, *Intentions in Communication*, pages 365–388. MIT Press, 1990.
- Y. He and S. Young. Spoken language understanding using the hidden vector state model. *Speech Communication*, 48(3-4):262–275, 2005.
- J. R. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as abduction. *Artificial Intelligence*, 63(1-2): 69–142, 1993.
- A. Lascarides and N. Asher. Agreement, disputes and commitment in dialogue. *Journal of Semantics*, 26(2): 109–158, 2009.
- O. Lemon and O. Pietquin. Machine learning for spoken dialogue systems. In *Interspeech*, 2007.
- K. E. Lochbaum. A collaborative planning model of intentional structure. *Computational Linguistics*, 24(4): 525–572, 1998.
- W. C. Mann and S. A. Thompson. Rhetorical structure theory: A framework for the analysis of texts. *International Pragmatics Association Papers in Pragmatics*, 1:79–105, 1987.
- M. Poesio and D. Traum. Towards an axiomatisation of dialogue acts. In J. Hulstijn and A. Nijholt, editors, *Proceedings of the Twente Workshop on the Formal Semantics and Pragmatics of Dialogue*. 1998.
- E. Rasmusen. *Games and Information: An Introduction to Game Theory*. Blackwell Publishing, 2007.
- W. Wahlster, editor. *Verbmobil: Foundations of Speech-to-Speech Translation*. Springer, 2000.

Appendix A : Treatment of an example

We illustrate in this section how our rules work on an example. Since this dialogue was also evaluated by our judges (cf section 7), we give where relevant some details on those annotations. The example is as follows:

- (6) π_1 . A: so, I guess we should have another meeting
 π_2 . A: how long do you think it should be for.
 π_3 . B: well, I think we have quite a bit to talk about.
 π_4 . B: maybe, two hours?
 π_5 . B: how does that sound.
 π_6 . A: deadly,
 π_7 . A: but, let us do it anyways.
 π_8 . B: okay, do you have any time next week?
 π_9 . B: I have got, afternoons on Tuesday and Thursday.
 π_{10} . A: I am out of Tuesday Wednesday Thursday,
 π_{11} . A: so, how about Monday or Friday

Table 4 is the DSDRS associated with (6). $Relation(\pi_i, [\pi_j - \pi_k])$ indicates that a rhetorical relation holds between the segment π_i and a segment consisting of $\pi_j, \pi_{j+1}, \dots, \pi_k$

π_1 provides an atomic preference. We apply the rule 1 and so $Commit_A(\pi_1, \mathcal{D}\mathcal{N}_A)$ adds the description $\mathcal{D}\mathcal{N}_A \models M \succ \overline{M}(CPT(M))$ where M means Meet.

π_2 We have $Q-Elab(\pi_1, \pi_2)$. A continues to commit to M in π_2 and no new preferences are introduced by π_2 . We apply rule 9, which makes the $P(\pi_2)$ description the same as $P(\pi_1)$'s.

π_3 is linked to π_2 with QAP . B accepts A's preference and we apply the rule 11 since π_2 is a *wh*-question. Thus $Commit_B(Elab_B(\pi_2, \pi_3), \mathcal{D}\mathcal{N}_B)$ adds the description $\mathcal{D}\mathcal{N}_B \models M \succ \overline{M}(CPT(M))$. It is interesting to note that some judges consider that agent's utterance in π_3 indicates a preference towards "talking a long time" while other judges consider, as our method predicts, that this segment does not convey any preference.

π_4 is linked to π_3 by $Q-Elab$. B commits to a new preference. We apply rule 9, rule 8 and then rule 8.a. The preference on the hour is now dependent on the preference on meeting; i.e., $\mathcal{D}\mathcal{N}_B \models M : 2h \succ \overline{2h}(CPT(2h))$, where the variable $2h$ means two hours.

π_5 is related to π_4 with the $Q-Cont$ relation. We then follow the same rule as the continued relation,

namely $Q-Elab$. We apply rule 9 which does not change the CP-net description of B because π_5 does not convey any preference.

π_6 is related to π_5 with QAP relation. In this case, it's not clear what is the QAP target and so we apply rule 12: we wait to learn more and we do not change B 's CP-net description.

All the Judges indicated that segments π_5 and π_6 are ambiguous and therefore hesitated to say if they commit to preferences. For example in π_6 , do we have a preference for meeting more than 2 hours or less than 2 hours? This indecision is compatible with the predictions of rule 12.

π_7 A accepts B's preference. We apply rule 9 and then rule 8 to obtain:

$$\begin{aligned} \mathcal{D}\mathcal{N}_A &\models M \succ \overline{M}(CPT(M)), \\ \mathcal{D}\mathcal{N}_A &\models M : 2h \succ \overline{2h}(CPT(2h)). \end{aligned}$$

π_8 is linked to π_7 by $Q-Elab$. B introduces a new preference for meeting next week.

We apply rule 9 and then 8 to obtain:

$$\begin{aligned} \mathcal{D}\mathcal{N}_B &\models M \succ \overline{M}(CPT(M)), \\ \mathcal{D}\mathcal{N}_B &\models M : 2h \succ \overline{2h}(CPT(2h)), \\ \mathcal{D}\mathcal{N}_B &\models M \wedge 2h : NW \succ \overline{NW}(CPT(NW)) \text{ where the variable } NW \text{ means next week.} \end{aligned}$$

π_9 is linked to π_8 by $Plan-Elab$. π_9 expresses commitments to preference that already involve a CP-net description. B introduces three preferences: one for meeting on Tuesday, the other for meeting on Thursday and given the conjunction of preferences $Tues \wedge Thurs$, one for time afternoon (of Tuesday and Thursday). That is, $((\diamond(Tues) \wedge \diamond(Thurs)) \mapsto Aft)(P(\pi_9))$. We apply the equivalence (3) and obtain :

$$(\diamond(Tues) \vee \diamond(Thurs)) \rightarrow Aft)(P(\pi_9)).$$

Then, we apply rules 8.g, 8.b and 8.d. The CP-net description of B is thus updated as follows:

$$\begin{aligned} \mathcal{D}\mathcal{N}_B &\models M \wedge 2h \wedge NW \wedge \overline{Tues} : Thurs \succ \overline{Thurs}(CPT(Thurs)), \\ \mathcal{D}\mathcal{N}_B &\models M \wedge 2h \wedge NW \wedge Tues : Thurs \sim \overline{Thurs}(CPT(Thurs)), \\ \mathcal{D}\mathcal{N}_B &\models M \wedge 2h \wedge NW \wedge \overline{Thurs} : Tues \succ \overline{Tues}(CPT(Tues)), \\ \mathcal{D}\mathcal{N}_B &\models M \wedge 2h \wedge NW \wedge Thurs : Tues \sim \overline{Tues}(CPT(Tues)), \\ \mathcal{D}\mathcal{N}_B &\models M \wedge 2h \wedge NW \wedge (Thurs \vee Tues) : Aft \succ \overline{Aft}(CPT(Aft)). \end{aligned}$$

Most judges express here a preference ranking over outcomes. For instance, if B elaborates by adding the preference "I have got Monday morning too" (as it is in the test corpus), some consider the ranking "(Tuesday or Thursday afternoons) \succ (Monday

Turn	A's SDRS	B's SDRS
1	$\pi_{1A} : Q\text{-Elab}(\pi_1, \pi_2)$	\emptyset
2	π_{1A} : is the same as in turn 1	$\pi_{2B} : Q\text{-Elab}(\pi_1, [\pi_2 - \pi_5]) \wedge QAP(\pi_2, [\pi_3 - \pi_5]) \wedge$ $Q\text{-Elab}(\pi_3, \pi)$ $\pi : Q\text{-Cont}(\pi_4, \pi_5)$
3	$\pi_{3A} : Q\text{-Elab}(\pi_1, [\pi_2 - \pi_7]) \wedge QAP(\pi_2, [\pi_3 - \pi_7]) \wedge$ $Q\text{-Elab}(\pi_3, [\pi_4, \pi_7]) \wedge QAP(\pi, \pi')$ $\pi : Q\text{-Cont}(\pi_4, \pi_5), \pi' : \text{Contrast}(\pi_6, \pi_7)$	π_{2B} : is the same as in turn 2
4	π_{3A} : is the same as in turn 3	$\pi_{4B} : Q\text{-Elab}(\pi_1, [\pi_2 - \pi_9]) \wedge QAP(\pi_2, [\pi_3 - \pi_9]) \wedge$ $Q\text{-Elab}(\pi_3, [\pi_4 - \pi_9]) \wedge QAP(\pi, [\pi_6 - \pi_9]) \wedge$ $Q\text{-Elab}(\pi', \pi'')$ $\pi : Q\text{-Cont}(\pi_4, \pi_5), \pi' : \text{Contrast}(\pi_6, \pi_7)$ $\pi'' : \text{Plan-Elab}(\pi_8, \pi_9)$
5	$\pi_{5A} : Q\text{-Elab}(\pi_1, [\pi_2 - \pi_{11}]) \wedge QAP(\pi_2, [\pi_3 - \pi_{11}]) \wedge$ $Q\text{-Elab}(\pi_3, [\pi_4 - \pi_{11}]) \wedge QAP(\pi, [\pi_6 - \pi_{11}]) \wedge$ $Q\text{-Elab}(\pi', [\pi_8 - \pi_{11}]) \wedge QAP(\pi'', \pi''')$ $\pi : Q\text{-Cont}(\pi_4, \pi_5), \pi' : \text{Contrast}(\pi_6, \pi_7)$ $\pi'' : \text{Plan-Elab}(\pi_8, \pi_9), \pi''' : Q\text{-Elab}(\pi_{10}, \pi_{11})$	π_{4B} : is the same as in turn 4

Table 4: The DSDRS for Dialogue (6).

morning) \succ (other days)", while others consider the ranking "(Tuesday or Thursday afternoon) or (Monday morning) \succ (other days)". We did not treat such preference ranking.

π_{10} is related to π_9 by QAP where A answers no to B 's question asked in π_8 . We apply rule 10 (no). Since $\overline{\text{Tues\&Weds\&Thurs}}(P(\pi_{10}))$ is not consistent with $((\diamond(\text{Tues}) \wedge \diamond(\text{Thurs})) \mapsto \text{Afi})(P(\pi_9))$, we apply $\text{Commit}_A(\text{Correction}(\pi_9, \pi_{10}), \mathcal{D}\mathcal{X}_A)$, which adds the preference $\overline{\text{Weds}}$ to A 's description and then the rule 13 where $\overline{\text{Tues}}$ and $\overline{\text{Thurs}}$ are respectively replaced by $\overline{\text{Tues}}$ and $\overline{\text{Thurs}}$:

$$\begin{aligned} \mathcal{D}\mathcal{X}_A &\models M \wedge 2h \wedge NW : \overline{\text{Tues}} \succ \text{Tues}(CPT(\text{Tues})), \\ \mathcal{D}\mathcal{X}_A &\models M \wedge 2h \wedge NW : \overline{\text{Thurs}} \succ \text{Thurs}(CPT(\text{Thurs})), \\ \mathcal{D}\mathcal{X}_A &\models M \wedge 2h \wedge NW : \overline{\text{Weds}} \succ \text{Weds}(CPT(\text{Weds})). \end{aligned}$$

π_{11} Finally, this segment is linked to π_{10} with $Q\text{-Elab}$ where $\text{Mond} \nabla \text{Fri}(P(\pi_{11}))$. We apply rules 9 and 8.b and update A 's CP-net description as follows:

$$\begin{aligned} \mathcal{D}\mathcal{X}_A &\models M \wedge 2h \wedge NW \wedge \overline{\text{Tues}} \wedge \overline{\text{Thurs}} \wedge \overline{\text{Weds}} \wedge \overline{\text{Fri}} : \text{Mond} \succ \text{Mond}(CPT(\text{Mond})), \\ \mathcal{D}\mathcal{X}_A &\models M \wedge 2h \wedge NW \wedge \overline{\text{Tues}} \wedge \overline{\text{Thurs}} \wedge \overline{\text{Weds}} \wedge \text{Fri} : \text{Mond} \sim \text{Mond}(CPT(\text{Mond})), \\ \mathcal{D}\mathcal{X}_A &\models M \wedge 2h \wedge NW \wedge \overline{\text{Tues}} \wedge \overline{\text{Thurs}} \wedge \overline{\text{Weds}} \wedge \overline{\text{Mond}} : \text{Fri} \succ \text{Fri}(CPT(\text{Fri})), \\ \mathcal{D}\mathcal{X}_A &\models M \wedge 2h \wedge NW \wedge \overline{\text{Tues}} \wedge \overline{\text{Thurs}} \wedge \overline{\text{Weds}} \wedge \overline{\text{Mond}} : \text{Fri} \sim \text{Fri}(CPT(\text{Fri})). \end{aligned}$$

The evaluation of this dialogue also reveals to what extent naive annotators reason with binary (*Monday preferred to not Monday*) or multi-valued variables (*Monday preferred to Tuesday*). Most judges use multi-valued variables to express the preference extracted from an EDU, and the way in which our method exploits domain knowledge to yield the minimal CP-net satisfying the description reflects this. In addition, some judges use a small set of variables (for example the variable *time of meeting* that groups together the notion of week, day, hours, etc.) while others use a distinct variable for each preference. Finally, we also noticed that judges do not describe the same preference dependencies. For example, in:

(7) We could have lunch together and then have the meeting from one to three?

some consider that the preference on having lunch is independent from the preference on the meeting (in this case, they consider that the preference on the period one to three is independent from the preference on meeting) while others consider that the two preferences are dependent.