From *Proceedings of the 15th International Congress on Cybernetics,* Namur (Belgium), August 1998

Belief Systems and Plans For Communication

Massimiliano Garagnani[1]
Department of Computer Science
University of Durham,
Durham, DH1 3LE (UK)

Abstract

Communication between rational agents can be seen as an exchange of internal *concepts* expressed using a shared *language.* Most of the communication acts between humans are performed through Natural Language (NL), and the *meaning* of the expressions of the language adopted is assumed to be known and properly understood by all of the participants. This suggests that the process of communication using NL requires the source-agent ('speaker') to hypothesise a model of its audience's ('hearer') *knowledge* and *beliefs*.

Moreover, the initial reasons which originate the *intention* (or need) to communicate (and which may or may not be included in the message) might consist of more abstract *perlocutionary* goals, such as the induction of a specific reasoning process (inference) within the hearer's mind producing the change of one or more of his/her beliefs. Since these goals play an important role in the process of determining the content and the structure of a discourse, it becomes necessary for the speaker not only to hypothesise and maintain a model of the hearer's beliefs, but also to plan the structure of the message according to the *effects* that the utterances will have upon them.

Whilst research in modal logics has produced many theoretical results on the subject of belief systems, the applications of these theories to other domains (like beliefs communication in NL) and the issues related to their actual implementation, such as belief grounding, have been, in a sense, neglected.

In this paper I introduce a *natural language oriented* model of beliefs which uses the concept of 'event' as fundamental unit for the knowledge representation. The basic features of the model defined include a qualitative degrees-of-belief approach to the representation of *uncertain* beliefs, two-level nested *mutual belief* expressions, and *grounding* of each belief through *rationality* and *experience.*

Secondly, taking this model as a starting point, I formalise a general Theoretical Framework for belief systems representation which embodies an inference-rules Boolean-functions based reasoning mechanism for beliefs extension and proving. This general model represents the knowledge on two different *levels of abstraction*, separating between *implicit* and *explicit* information. Each of these two levels contains propositions expressed using a distinct language of beliefs: the 'outer' language, adopted to model the implicit knowledge, and the 'inner' language, used to specify explicit beliefs and constituting a proper subset of the former.

Finally, I illustrate the integration of the specific model implemented - which meets the requirements of the above mentioned theoretical paradigm - within a *planning* system, in which the set of beliefs held by the system represents the current *mental state* of the speaker. Accordingly, any solution found by the planner will be composed of a sequence of actions (*operator instances*) that transforms the initial state into a final mental state in which the assigned perlocutionary goal is achieved. Hence, the system resulting from this integration constitutes an automatic generator of *plans* for *NL communication of beliefs* among agents which may be driven by perlocutionary intentions.

[1] E-mail: Massimiliano.Garagnani@durham.ac.uk.

1 INTRODUCTION

The increasing complexity of multiagent systems applications, in which heterogeneous agents, with different goals, knowledges and beliefs, coexist and interact in a dynamic environment, along with the need for the progressive development of a user-friendly information society, underline the necessity of implementing systems with sophisticated and *natural-language* oriented communicational skills.

The ability of an agent to perform complex communications, such as explanations or persuasive discourses, requires to maintain an internal model of the 'Hearer' beliefs and knowledge, which must be taken into account during the process of construction of a communication plan to determine the structure and content of the message [2][3][7].

Therefore, it appears necessary to design agents endowed with a special-purpose communication system, able to produce natural-language oriented communication plans based on beliefs models which contain a representation of both the Speaker's and Hearer's knowledge and beliefs[4].

This paper describes and formalises a beliefs system which has been defined to meet these requirements, and explains how it has been successfully used to build plans for persuasive communication.

2 THE BELIEF SYSTEM

2.1 Events and beliefs

The system of beliefs designed to be used by an agent to plan complex communication consists of an evolving collection of semantic units called 'events'. The conceptual unit of event is commonly used in semantic networks to express sentences, and represents the equivalent of a predicate logic proposition of the type *Action(subject, object)* (see also [6] [1]).

In the model of beliefs implemented, the event formalism is recursive, allowing events to be subjects or objects of other events. For example, if 'e1' is an event, "Bel(Agent, e1)" is the belief equivalent to the proposition "Agent believes that e1".

Since the model defined allows for the presence of uncertainty by adopting a simple qualitative tripartite degrees-of-beliefs approach, as described in [1], every event 'e' in the knowledge will be regarded as being in one and only one of the three following categories: c1) *Believe*(A,e), abbreviated with ABel(e), c2) *Believe*(A,¬e), abbreviated with ABel(¬e), and c3) *Undecided*(A,e), abbreviated with AUnd(e). Notice that the subject of the beliefs is the agent 'A', whereas '¬e' refers to the negation of the event 'e'.

2.2 Support and grounding of beliefs

In a rational belief system, every belief holding must have a rational justification. The model admits two possible ways in which a belief ABel(e) can be rationally justified: either i) 'e' is a *consequence* of A's reasoning about others beliefs, or ii) 'e' is a direct *experience* of A's physical perceptions (e.g. e='A eats an apple'). The latter kind of belief is considered unquestionable and will never be abandoned by the agent, under the assumption that A believes to be perfectly reliable and unbiased about the memories concerning his/her perceptions of the physical reality.

The two notations adopted to represent these two aspects - reasoning and experience - of belief justification for an event 'e' are, respectively, *Support(e',e)* and *Real(A,e)*, abbreviated with Sup(e',e) and AReal(e).

The expression Sup(e’,e) means that the belief in *e’* is a sufficient reason to believe *e*; notice that the belief ABel(Sup(e’,e)) constitutes an event which must be justified. The expression AReal(e) indicates that the (object) event *e* is a *real experience* for the (subject) agent A; in this case, the belief ABel(e) would not require further support, nor would the belief ABel(AReal(e)).

Finally, it is necessary to identify, in the model of an agent A, those events which (s)he believes to be *unknown* by another agent ‘B’. For this aim, I introduced the modal predicate *Unknown(B,e)* (abbreviated with BUnk(e)), representing A’s belief in the fact that the event *e* is unknown to B, i.e. it has never been ‘conceived’ as a thought by B’s mind.

2.3 Formalisation of the language

The syntax defining the well-formed expressions (WFE) used to specify the beliefs set is determined by the following BNF production rules:

<WFE> ::= <Pred> (<A>) | NOT (<Pred> (<A>))
<Pred> ::= SBEL | SUND | HBEL | HUND | HUNK
<A> ::= <E> | NOT (<E>)
<E> ::= SREAL (<Es>) | HREAL (<Es>) | <Es>
<Es> ::= SUP (<En>, <En>) | <Ei>
<En> ::= <Es> | NOT (<Es>)
<Ei> ::= E_0 | E_1 | E_2 | E_3 | .... | E_n

Notice that ‘S’ and ‘H’ stand, respectively, for ‘Speaker’ and ‘Hearer’: this is a necessary distinction, as the model is meant to be used as a base to plan communication between two agents. Notice also that this syntax admits two levels of belief nesting: in fact, every expression generated by <WFE> can be thought of as having the predicate ‘SBEL’ as prefix, as representing one of the Speaker’s beliefs. For example, HBEL(E_2) should be interpreted as SBEL(HBEL(E_2)). Finally, it should also be pointed out that each of the two arguments of a SUP( ) relationship can consist of another SUP( ) relationship, allowing the generation of expressions with an arbitrarily high number of nested supports.

However, this language is not completely suitable to be used directly to specify the current set of beliefs holding. For example, many expressions of this language are ‘redundant’, i.e. have an equivalent ‘meaning’ (like SUND(e) and SUND(NOT(e)) and should always be treated as if representing a single belief, causing duplication of operations and redundant checks at run time. Moreover, considering that each event ‘e’ present in the knowledge must belong to one and only one of the three categories c1) - c3), and that every SBEL(e) proposition must be fully justified, it is reasonable to think that the largest portion of events included in the knowledge will be of the third type (c3), i.e. classified as uncertain information. This suggests that, instead of including explicitly the expressions ‘SUND(e)’ and ‘SUND(NOT(e))’ in the set of beliefs, it would be more convenient to assume that all the events not mentioned in c1) or c2) must belong, by exclusion, to c3). In other words, SUND(e) and SUND(NOT(e)) can be assumed to hold by *default* in absence of any specific ‘attitude’ of the speaker towards the event ‘e’ (i.e. when neither the belief SBEL(e) nor SBEL(NOT(e)) is present).

This and others considerations (explained also in [1]) have led to the introduction of a *belief prover* algorithm to act as *interface* between the set of ‘outer’ expressions generable by <WFE> and the set of ‘inner’ beliefs actually stated. This algorithm is designed to accept in input any <WFE> expression and calculate its truth-value according to the presence or absence of associated belief(s) in a smaller and unequivocal set of ‘inner’ expressions. This simpler set of beliefs will be specified using an 'inner' language, corresponding to the subset of <WFE> obtained removing all the redundant expressions and default beliefs.

In the following section, I shall define the theoretical model for belief systems which generalises this idea and can be applied to any formal language for knowledge representation.

3 THEORETICAL FRAMEWORK

3.1 Boolean evaluations and functions

Let us consider a language *L* consisting of a set of 'well formed formulas', or expressions, defined over a finite alphabet *F*. *L* does not have to be finite.

Definition A *Boolean Evaluation* of a language *L* is a function B: $L \rightarrow \{\text{True, False}\}$ such that $\forall$ $e \in L$, the value of $B(e)$ is decidable.

To assign a boolean evaluation is equivalent to identify the current set of beliefs held by the system, which is obviously defined as the set $S = \{p \in L \mid B(p) = True\}$.

Definition A *Boolean Interpretation* of a language *L* is a set of boolean evaluations of *L*.

Since every boolean evaluation of *L* identifies a set of 'True' beliefs, a boolean interpretation of a language can be used to specify the collection of permitted sets of beliefs which satisfy certain conditions, such as coherence or completeness.

Let's consider a finite subset of expressions $S=\{e_1,e_2,...,e_n\} \subseteq L$, and the collection of terms $B(e_1)$, $B(e_2)$, ..., $B(e_n)$ obtained applying a boolean evaluation $B(\ )$ to this set.

Definition A *Boolean Function* of a language *L* defined on $S = \{e_1,e_2,...,e_n\} \subseteq L$ is a *logical expression* containing the terms $B(e_1),...,B(e_n)$ which is built using the connectives $\wedge,\vee$ and the unary operator '$\neg$' (logical negation), where $B(\ )$ represents a boolean evaluation of *L*.

For example, if $S=\{e_1,e_2\} \subset L = \{e_1,e_2,....,e_n,.....\}$, $F(e_1,e_2) = \neg (B(e_1) \vee B(e_2))$ is a boolean function of *L* defined on S, the truth value of which depends on the boolean values $B(e_1)$, $B(e_2)$.

3.2 Inference rules and extension

Considering that a set of beliefs can be thought of as a subset of a language *L*, it is useful to introduce a formalisation of what could be called an 'inference rule', which produces new beliefs as a result of reasoning based on beliefs already holding.

Definition An *Inference Rule R* for a language *L* is an association between a non-empty set of expressions (premises) $S_{left} \subseteq L$ and a non-empty set of expressions (consequences) $S_{right} \subseteq L$, written as

$$R) \qquad S_{left} \ \vdash \ S_{right}$$

The intuitive meaning of this definition is that if all the premises specified in S_{left} are contained in the belief set, then also the set of consequences S_{right} should be added to the set. This operation can be defined as the 'extension' of the beliefs set:

Definition Given a set *IR* of inference rules for a language *L* and a set S $\subseteq$ *L*, the *extension* '$Ext_{IR}(S)$' of S using *IR* consists of the set $S_{IR} \subseteq L$ obtained applying - recursively - all the inference rules contained in *IR* to the set S.

3.3 The Belief Prover Algorithm

Now that all the necessary tools have been introduced, it is possible to define the structure which constitutes a general model for belief systems. Let *Lo* be an (outer) language, and $Li \subset Lo$ an (inner) language properly contained in *Lo*. Let also *J* be a boolean interpretation of *Li*.

Definition An *Interpretation* of *Lo on Li* is a (calculable) function which associates every outer expression $p \in Lo$ to a boolean function of *Li*.

An *algorithm* which, given an outer expression $p \in Lo$, calculates and returns the truth of the associated boolean function - determined using the set $S \subseteq Li$ of (inner) beliefs currently true - will be said to *calculate* an interpretation of *Lo* on *Li*, and will be referred, in the rest of this paper, as a *Belief Prover (BP)* for *Lo*. The value returned by a *BP function* can be defined as *BP(p,*S*)*, where $BP : Lo \times 2\text{^}Li \rightarrow \{\text{True, False}\}$, and $2\text{^}Li$ represents the set of subsets of *Li*.

Theorem 1 The function *BP( )* calculates a boolean evaluation of *Lo* based on a boolean evaluation of *Li*.

Proof By definition, *BP(p,*S) is a (decidable) function which returns 'True' or 'False' depending on the given outer expression *p*, and therefore constitutes a boolean evaluation of *Lo*. On the other hand, the set 'S' - containing the (inner) beliefs currently held - identifies unequivocally a boolean evaluation of *Li*, and is used to deduce the truth value of *p* from the boolean function associated to it.

Finally, it is necessary to explain the role that the inference rules play in this framework. The set $S \subseteq Li$ of currently holding beliefs must be one of the admitted boolean evaluations in *J*. If we effect the extension of S using a set *IR* of inference rules, we shall obtain the extended set of beliefs $S_{IR} \subseteq Li$ which must correspond to another admitted boolean evaluation. S_{IR} will then be used by the *BP* algorithm as the current evaluation (set of beliefs holding) to evaluate an outer expression. Therefore, the set *IR* of inferences rules for the inner language *Li* should be defined taking into account the given boolean interpretation *J*, in order to avoid the violation of the specific properties (e.g. of coherence) which the set of beliefs must satisfy.

Summarising, a *Belief Prover* algorithm for *Lo* calculating an interpretation of *Lo* on *Li* ($\subset$ *Lo)* will consist of the following steps:

Let S = the current 'core'-set of beliefs $\subset Li$, and *IR* = the set of inference rules for *Li*;
Let *p* = the outer language expression to be evaluated.

1. Calculates $S_{IR} = Ext_{IR}(S)$ using the given set *IR* of inference rules.
2. Transform *p* into the corresponding boolean function $A(B(e_i),B(e_j),...,B(e_k))$ of inner expressions $e_i,e_j,...,e_k$.
3. Determine the current boolean values $B(e_i),..,B(e_k)$ simply checking for the *presence* of each 'e_i', 'e_j' , ..., 'e_k' in S_{IR}.
4. Use these values to calculate the truth value of *A*. Return this as result.

4 THE SYSTEM

4.1 Specific characteristics

The beliefs system, implemented according to the described theoretical paradigm, can be specified as follows:

- *Lo* =<WFE>;
- *Li* = <WFF>, where

  <WFF>::= <A> | HBEL (<A>) | HUND (<Eh>)
  <Eh> ::= SR(<Es>) | <Es>

- Inference rules set: *IR* = {I1,I2,I3,I4,I5}, where

  I1) SREAL(*e*) |- e
  I2) HREAL(*e*) |- *e*, HBEL(HREAL(*e*))
  I3) NOT(HREAL(*e*)) |- HBEL(NOT(HREAL(*e*))
  I4) HBEL(SREAL(*e*)) |- HBEL(*e*)
  I5) HBEL(HREAL(*e*)) |- HBEL(*e*)

  and '*e*' is a variable which can be substituted with any legal expression generated by <Es>.

- Boolean functions: let *x* be an expression generated by <WFE>. The boolean function associated to *x* is defined by *check(x')*, where *x'* is obtained from *x* by applying, when possible, the following syntactical transformations:

  a) SUND (NOT (*e*)) → SUND(*e*)
  b) HUND(NOT (*e*)) → HUND(*e*)
  c) SBEL (*a*) → *a*

  The function *check( )* is defined as follows:

$$
check(x) = \begin{cases}
B(x) & \text{if} \quad x = \mathrm{Ei} \,|\, \mathrm{SR}(e) \,|\, \mathrm{HR}(e) \,|\, \mathrm{NOT}(\mathrm{SUP}(e,e)) \,| \\
 & \quad \mathrm{SUP}(e,e) \,|\, \mathrm{NOT}(\mathrm{HR}(e)) \,|\, \mathrm{NOT}(\mathrm{Ei}); \\
B(\mathrm{NOT}(\mathrm{SR}(e)) \vee \neg B(\mathrm{SR}(e)) & \text{if} \quad x = \mathrm{NOT}(\mathrm{SR}(e)); \\
\neg\, check(p) & \text{if} \quad x = \mathrm{NOT}(p); \\
B(\mathrm{HBEL}(a)) & \text{if} \quad x = \mathrm{HBEL}(a); \\
B(\mathrm{HUND}(e)) & \text{if} \quad x = \mathrm{HUND}(e); \\
\neg (B(e) \vee B(\mathrm{NOT}(e))) & \text{if} \quad x = \mathrm{SUND}(e); \\
\neg (\, B(\mathrm{HBEL}(e)) \vee B(\mathrm{HUND}(e)) & \\
\quad \vee B(\mathrm{HBEL}(\mathrm{NOT}(e)))\,) & \text{if} \quad x = \mathrm{HUNK}(e);
\end{cases}
$$

  where 'SR' and 'HR' are abbreviations for 'SREAL' and 'HREAL', and *B( )* is the current boolean evaluation of *Li*, which returns 'True' iff its argument belongs to the current (extended) set of beliefs S_{IR}.

4.2 Integration of the model in a planning system

The model of beliefs implemented has been designed to be integrated into a planner to build a system for automatic generation of plans for persuasive communication. The integration has been realised by transforming the BP algorithm into a set of inference rules for the outer language *Lo*. These rules extend every boolean evaluation of *Li* into the corresponding boolean evaluation of *Lo* calculated by the BP (in virtue of Theorem 1). In other words, given a set S of (inner) beliefs $\subseteq$ *Li* which represents the current mental state, S is extended into a set S' of (outer) beliefs containing *all and only* the expressions of *Lo* which would be evaluated 'True' by the BP.
This extension has been realised as a *pre-process* performed off-line on both the initial state and the set of operators to be used by the planner. For example, the simple (inner) state S = {SR(E_1), HUND(E_1)} will be extended to the following set of (outer) beliefs :

S' = {SR(E1), HUND(E1), E1, SUND-HR(E1), SUND-NOT-HR(E1), HUNK-HR(E1), HUNK-NOT-SR(E1), HUNK-NOT-HR(E1), HUNK-SR(E1), SBEL-SR(E1), NOT-SUND-NOT-SR(E1), NOT-SBEL-NOT-SR(E1), NOT-SUND-SR(E1), HUND-NOT(E1), NOT-HUNK-NOT(E1), NOT-HBEL-NOT(E1), NOT-HUNK(E1), NOT-HBEL(E1), SBEL(E1), NOT-SUND-NOT(E1), NOT-SBEL-NOT(E1), NOT-SUND(E1), NOT-SBEL-HR(E1), NOT-SBEL-NOT-HR(E1), NOT-HBEL-HR(E1), NOT-HUND-HR(E1), NOT-HUND-NOT-HR(E1), NOT-HBEL-NOT-HR(E1), NOT-HBEL-NOT-SR(E1), NOT-HUND-NOT-SR(E1), NOT-HUND-SR(E1), NOT-HBEL-SR(E1)}[2]

The advantage of this approach lies in its general validity: any belief prover implemented according to the theoretical paradigm described can be transformed into a set of inference rules for the specific outer language, and, more importantly, this transformation does not require *any modification* of the planner's structure or code: the initial state and the operators can be extended only once and for all, before the actual planning process take place. This offers the opportunity to integrate a BP into a wide variety of planners. Practical experiments have been carried out using, for example, the 'IPP' planning system [5], which has produced communication plans with 8-10 steps in less than 3 seconds of total execution time.

5 CONCLUSIONS

Complex communication amongst agents requires a speaker to model the knowledge and beliefs of the audience, and to plan the content and structure of the message according to these information. This paper has introduced and formalised a natural-language oriented model for speaker's and hearer's beliefs, and has shown how it has been used in conjunction with a planner to produce complex plans for communication. The realisation of this system, able to plan persuasive 'monologues' (discourses), constitutes a basic step towards the development of systems for the automatic generation of *dialogues* in natural language.

6 BIBLIOGRAPHY

[1] Garagnani, M. (1997) "Belief Modelling for Discourse Plans", in *Proceedings of the 16th Workshop of the UK Planning and Scheduling Special Interest Group*, 1997.
[2] Grosz, B.J., Pollack, M.E., Sidner, C.L. (1989) "Discourse", in Posner, ed. *Foundations of Cognitive Science* (MIT Press), pp.437-468
[3] Grosz, B.J., Sidner, C.L., (1986) "Attention, intention, and the structure of discourse", *Computational Linguistics*, **12**(3):175-204.

[2] Notice that expressions with nested brackets have been transformed into single predicate expressions, e.g. SUND(NOT(HR(E1))) has become SUND-NOT-HR(E1).

[4] Grosz, B.J., Sidner, C.L. (1990) "Plans for discourse", in Cohen P., Morgan J., Pollack M.E., eds. *Intentions in Communication* (MIT Press, Cambridge, MA), pp.417-444.
[5] Koehler, J., Nebel, B., Hoffmann, J., Dimopoulos, Y. (1997) "Extending Planning Graphs to an ADL Subset", *Proceedings ECP*-97.
[6] Smith, M.H., Garigliano, R., Morgan, R. (1994) "Generation in the LOLITA system: an Engineering Approach", *Proceedings of the 7th International NL Generation Workshop*, Maine.
[7] Young, R.M., Moore, J.D., Pollack, M.E. (1994) "Towards a Principled Representation of discourse Plans", in *Proceedings of the Sixteenth Annual Meeting of the Cognitive Science Society*, Atlanta, GA.