Modeling credulousness and cooperativeness in a Logic Programming Framework

Paulo Quaresma and José Gabriel Lopes
{pq,gpl}@di.fct.unl.pt
Departamento de Informática,
Faculdade de Ciências e Tecnologia,
Universidade Nova de Lisboa,
2825 Monte da Caparica, Portugal

Abstract

We propose a logic programming framework that allows to model agents with different degrees of credulousness and cooperativeness. By credulousness we mean the characteristic that defines how an agent accepts the information that is conveyed by the different speech acts; by cooperativeness we mean the characteristic that defines how an agent relates his intentions with the other agents intentions.

In this framework each agent is modeled by a set of extended logic programming rules representing its mental state. These rules describe the agent behavior, attitudes (believes, intentions, and objectives), world knowledge, and temporal and reasoning procedures. The complete mental state is defined by the well founded model of the extended logic program that models the agent.

It will be shown how different behaviors can be modeled and some examples will be presented.

1 Introduction

In order to participate in dialogues, an agent needs the capability to model its mental state. Namely, it is necessary to represent the agent attitudes (believes, intentions, and objectives), world knowledge and temporal, reasoning and behavior rules. In this paper we propose a methodology that uses a logic programming framework to model agents. Agent models are defined as logic programs extended with explicit negation and the semantics of the programs is given by the well founded semantics of logic programs with explicit negation (from Pereira et al. [AP96; ADP95; Alf93]). The well founded semantics has a complete and sound top-down proof procedure with polynomial complexity and there is an implemented prototype ([DNP94]) which allows us to obtain experimental results.

In this paper we will show how some important behavior characteristics can be modeled by extended logic programming rules. Namely, it will be shown how credulousness and cooperativeness can be modeled.

Credulousness defines the way an agent accepts the information that is conveyed by the different speech acts. An agent may have several degrees of credulousness:

1. Always accept the new information;
2. Accept the new information if it is plausible;
3. Accept the new information only if it is not incompatible with the agent mental state.

In the first degree, the agent always accepts the new information and updates his mental state accordingly. Note that this update process may lead to a contradictory mental state, because the agent may had previous contradictory beliefs. In such a situation it is necessary to revise the agents mental state, terminating the oldest attitudes that supported the contradiction. The revising process is done through the use of the Contraction Removal Semantics of eXtended Logic Programs (CRSX). In the second degree of credulousness, the agent only accepts the new information it is possible to infer a sequence of hypothetical actions that leads to the specified state. This means that the agents only accept what is plausible in their world: "My car flies" is only accepted by an agent if that property may be supported by its model. Finally, in the third degree of credulousness, an agent accepts an information if it is not incompatible with his previous mental state.

Cooperativeness defines how an agent relates his intentions with the other agents intentions. Namely, it defines how an agent initiates a new intention based on the other agents inferred intentions. It can have several degrees:

1. Always accept the other agents intentions;
2. Accept the other agents intentions only if they are not contradictory with the agents model;
3. Accept the other agents intentions if it is possible to satisfy them.

In the first degree, the agent accepts the other agents intentions revising his model whenever is necessary. This means that the agent may terminates previous intentions if they are contradictory with the new ones: to open and to close a door. In the second degree of cooperativeness, the agent only accepts an intention if it is not contradictory with his own model. In the third degree of cooperativeness, the agent only accepts the intention if there exists a sequence of actions that leads to a state where it is possible to execute the intended action.

In a dialogue, after each sentence, an agent updates his model with the new information. This process is done through the update of the logic program with the facts that describe the events: identification of the time and speech acts associated with each event. After this process, the revised model is calculated using the CRSX.

In the next section, the logic programming framework is briefly described. In section 3 we present the agent modelling process, with a special focus on the capability to model agents with different levels of credulousness and cooperativeness. In section 4 the procedures to update and revise the agents mental state after each event are briefly described. In section 5 some examples illustrating the dialogue participation process are presented and, finally, in section 6 some conclusions and open problems are pointed out.

2 Logic programming framework

Logic programs extended with explicit negation are finite set of rules of the form

- $H \leftarrow B_1, \ldots, B_n$, not $C_1, \ldots$, not C_m $\quad (m \geq 0, n \geq 0)$

where H, B_1, ..., B_n, C_1,..., C_m are objective literals. An objective literal is an atom A or its explicit negation $\neg A$; *not* stands for negation by default; *not* L is a default literal. Literals are objective or default and $\neg\neg L \equiv L$.

The set of all ground objective literals of a program P designates the extended Herbrand base of P and it is represented by $\mathcal{H}(P)$. An interpretation I of an extended program P is represented by $T \cup$ *not* F, where T and F are disjoint subsets of $\mathcal{H}(P)$. Objective literals of T are true in I; objective literals of F are false by default in I; objective literals of $\mathcal{H}(P) - I$ are undefined in I. Moreover, if $\neg L \in T$ then $L \in F$.

An interpretation I of an extended logic program P is a partial stable model of P iff $\Phi_P(I) = I$ (see [AP96] for the definition of the Φ operator).

The well founded model of the program P is the F-least partial stable model of P. The well founded semantics of P is determined by the set of all partial stable models of P.

Pereira et al. ([AP96; ADP95]) showed that every non-contradictory program has a well founded model and they also presented a complete and sound top-down proof procedure for several classes of programs.

In their work, Pereira et al., proposed a revision process that restores consistency for contradictory programs, taking back assumptions of the truth value of negative literals. As it will be described in section 4, we also use this approach in order to revise the agents mental state.

2.1 Events

The agent modeling process must be able to deal with time and events. In fact, it is very important that agents have the capability to reason about their mental state at a given time point. They should also be able to change their mental state as a consequence of some external or internal events.

As a time formalism we propose a variation of the Event Calculus ([Sha89; Esh88; Mis91]) that allows events to have an identification and a duration. As a consequence events may occur simultaneously.

The predicate *holds_at* defining the properties that are true at a specific time is:

$$\begin{aligned}
holds\_at(P,T) &\leftarrow happens(E,T_i,T_f), & (1)\\
&\qquad initiates(E,T_P,P), &\\
&\qquad T_P < T, &\\
&\qquad persists(T_P,P,T). &\\
persists(T_P,P,T) &\leftarrow not\ clipped(T_P,P,T). & (2)\\
clipped(T_P,P,T) &\leftarrow happens(C,T_{ci},T_{cf}), & (3)\\
&\qquad terminates(C,T_C,P), &\\
&\qquad not\ out(T_C,T_P,T). &\\
out(T_C,T_P,T) &\leftarrow T \leq T_C. & (4)\\
out(T_C,T_P,T) &\leftarrow T_C < T_P. & (5)
\end{aligned}$$

The predicate $happens(E,T_i,T_f)$ means that the event E occurred between T_i and T_f; $initiates(E,T,P)$ means that the event E initiates P at time T; $terminates(E,T,P)$ means that the event E terminates P at time T; $persists(T_i,P,T)$ means that P persists since T_i until T (at least); $succeeds(E,T_i)$ means that the event E may occur at time T_i (its pre-conditions are satisfied).

Note that a property P is true at a time T ($holds\_at(P,T)$, if there is a previous event that initiates P and if P persists until T. P persists until T if it can not be proved by default the existence of another event that terminates P before the time T.

We need additional rules for the relation between not holding a property and holding its negation and we also need to define the relation between the two kinds of negation:

$$\neg \mathit{holds\_at}(P,T) \;\leftarrow\; \mathit{holds\_at}(\neg P,T). \tag{6}$$

$$\neg \mathit{holds\_at}(P,T) \;\leftarrow\; \mathit{not}\ \mathit{holds\_at}(P,T). \tag{7}$$

The predicates need to be related by some integrity rules:

1. Events can not initiate and terminate a property at the same time:

$$\leftarrow \mathit{initiates}(E,T,P),\ \mathit{terminates}(E,T,P). \tag{8}$$

2. Events can not initiate/terminate a property and its negation:

$$\leftarrow \mathit{initiates}(E,T,P), \mathit{initiates}(E,T,\neg P). \tag{9}$$

$$\leftarrow \mathit{terminates}(E,T,P), \mathit{terminates}(E,T,\neg P). \tag{10}$$

3. Events can not be associated to different time intervals:

$$\begin{aligned} \leftarrow \mathit{happens}(E,T_{1i},T_{if}), \\ \mathit{happens}(E,T_{2i},T_{2f}), \\ T_{1i} = T_{2i}, \\ \mathit{not}(T_{if} = T_{2f}). \end{aligned} \tag{11}$$

4. Events can not have a negative duration:

$$\leftarrow \mathit{happens}(E,T_i,T_f),\ \mathit{not}(T_i \leq T_f). \tag{12}$$

5. Events must have an associated action:

$$\begin{aligned} \leftarrow \mathit{happens}(E,T_i,T_f), \\ \mathit{not}(\mathit{act}(E,A)). \end{aligned} \tag{13}$$

6. Properties must be initiated by some event:

$$\begin{aligned} \leftarrow \mathit{holds\_at}(P,T), \\ \mathit{not}(\mathit{ev\_gen}(P,T)). \\ \mathit{ev\_gen}(P,T) \leftarrow \mathit{happens}(E,T_i,T_f), \\ \mathit{initiates}(E,T_p,P), \\ T_i \leq T_p \leq T, \\ \mathit{persists}(T_p,P,T). \end{aligned} \tag{14}$$

7. Events can not occur if the pre-conditions are not satisfied:

$$\leftarrow \mathit{happens}(E,T_i,T_f),\ \mathit{not}\ \mathit{succeeds}(E,T_i). \tag{15}$$

3 Agents mental state

In our proposal, agents are modeled by the well founded model of an extended logic program with the following structure:

1. Rationality rules (RR). These rules describe the relation between the different attitudes (beliefs, intentions, and objectives).
2. Behavior rules (BR). These rules define the agent's credulousness, cooperativeness, activity, and sincerity.
3. Actions description (Ac). These rules describe the actions that may be executed by the agent. In the domain of dialogues, these rules describe the speech acts, their pre-conditions and effects.
4. A temporal formalism (T). These are the rules presented in the previous section.
5. World knowledge (WK).These rules describe the agent's world knowledge: entities, taxonomies, ...

In this paper we will analyze only the first two structures: rationality rules and behavior rules.

3.1 Rationality rules

These rules define relations between agents' attitudes: beliefs (bel), objectives (ach), and intentions (int).

The main relations are (for related work see [Bra90; CL90a; CL90b; Per90]):

- Integrity

$$\begin{aligned} \bot \leftarrow\ & \mathit{holds\_at}(\mathit{bel}(A,P),T), \\ & \mathit{holds\_at}(\mathit{bel}(A,\neg P),T). \\ \bot \leftarrow\ & \mathit{holds\_at}(\mathit{ach}(A,P),T), \\ & \mathit{holds\_at}(\mathit{ach}(A,\neg P),T). \end{aligned}$$

- Consistency

$$\begin{aligned} \neg \mathit{holds\_at}(\mathit{bel}(A,\neg P),T) &\leftarrow \mathit{holds\_at}(\mathit{bel}(A,P),T). \\ \neg \mathit{holds\_at}(\mathit{ach}(A,\neg P),T) &\leftarrow \mathit{holds\_at}(\mathit{ach}(A,P),T). \end{aligned}$$

- Introspection

$$\begin{aligned} \bot \leftarrow \mathit{holds\_at}(\mathit{bel}(A,P),T), \\ \mathit{holds\_at}(\mathit{bel}(A,\neg \mathit{bel}(A,P)),T). \\ \bot \leftarrow \neg \mathit{holds\_at}(\mathit{bel}(A,P),T), \\ \mathit{holds\_at}(\mathit{bel}(A,\mathit{bel}(A,P)),T). \end{aligned}$$

- Necessity

$$\mathit{holds\_at}(\mathit{bel}(A,P),T) \;\leftarrow \mathit{holds\_at}(P,T).$$

3.2 Behavior rules

These rules allow the definition of the agent behavior. As pointed out previously, in this work we have considered only the credulousness and cooperativeness.

Credulousness

Credulousness defines how an agent accepts information from other agents.

In the first degree of credulousness, an agent believes everything he believes the other agents believe:

$$\begin{aligned} &holds\_at(bel(H,P),T) \leftarrow \\ &\quad holds\_at(bel(H,bel(S,P)),T). \end{aligned} \tag{16}$$

The credulousness property is also connected with the description of the speech acts. In fact, speech acts initiate some beliefs in the hearers of those acts.

In this paper, we will show only the effect of the *inform* speech act for sincere agents:

$$\begin{aligned} &initiates(E,T_f,bel(H,bel(S,P))) \leftarrow \\ &\quad act(E,inform(S,H,P)), \\ &\quad happens(E,T_i,T_f). \end{aligned} \tag{17}$$

Note that in this degree of credulousness, it is possible to have a contradiction:

- A: The door is opened.
- B: OK.
- A: The door is closed.

After the first utterance, agent B believes that the door is opened:

$$holds\_at(bel(b,door\_opened),t_0).$$

(using the rule for the inform speech act 17 and the credulousness rule 16).

After the third utterance, there is a contradiction:

$$holds\_at(bel(b,door\_opened),t_1).$$
$$holds\_at(bel(b,door\_closed),t_1).$$

(we have assumed an integrity constraint that defines that is contradictory to believe simultaneously that a door is opened and closed).

In a contradictory state, the model must be revised and it should be selected a preferred non-contradictory model. The selection process is done through the use of preference rules (for instance, preferring the models which revise the oldest/newest attitude).

In the second degree of credulousness, an agent believes in a property he believes the other agents believe if that property is plausible:

$$\begin{aligned} &holds\_at(bel(H,P),T) \leftarrow \\ &\quad holds\_at(bel(H,bel(S,P)),T), \\ &\quad holds\_at(bel(H,possible(P)),T). \end{aligned} \tag{18}$$

A property P is possible if there is a hypothetical sequence of actions that leads to the state P. This process is verified by the construction of a hypothetical world where property P holds. This is done adding new integrity constraints that force the property to hold in the future (note that in our framework events and actions may be abduced in order to satisfy integrity constraints):

$$IC'(t) = IC(t) \cup \{holds\_at(P,t') \leftarrow, t < t'\}$$

If the hypothetical world is non-contradictory, then the property is plausible and it may be accepted.

The third degree of credulousness accepts the new information only if it the new model is not contradictory, i. e., there exists a well founded model of the new logic program.

Cooperativeness

This property defines how intentions and objectives are transferred between agents.

In the first degree of cooperativeness, an agent always accepts intentions and objectives:

$$\begin{aligned} &holds\_a(int(H,A),T) \leftarrow \\ &\quad holds\_at(bel(H,int(S,A)),T). \end{aligned} \tag{19}$$

$$\begin{aligned} &holds\_a(ach(H,P),T) \leftarrow \\ &\quad holds\_at(bel(H,ach(S,P)),T). \end{aligned} \tag{20}$$

In the second degree, the agent only accepts intentions and objectives if they are not contradictory with the previous model, i. e., if there exists a well founded model of the new logic program.

In the third degree of cooperativeness, the agents accepts intentions if he believes it is possible to satisfy them:

$$\begin{aligned} &holds\_a(int(H,A),T) \leftarrow \\ &\quad holds\_at(bel(H,int(S,A)),T), \\ &\quad holds\_at(bel(H,possible\_action(A)),T). \end{aligned} \tag{21}$$

$$\begin{aligned} &holds\_a(ach(H,P),T) \leftarrow \\ &\quad holds\_at(bel(H,ach(S,P)),T), \\ &\quad holds\_at(bel(H,possible(P),T). \end{aligned} \tag{22}$$

An objective is possible if there exists a hypothetical sequence of actions that leads to that state (see definition in the previous subsection).

An action a is possible if there is a hypothetical state where that action can be executed. This is done adding new integrity constraints that force the action to occur in the future:

$$IC'(t) = IC(t) \cup \{happens(e', t'_i, t'_f), act(e', a), t < t'_i \leq t'_f\}$$

If the hypothetical world is non-contradictory, then the action is possible.

4 Updating and revising an agent mental state

The agent mental state, as it was defined in the previous sections, must be updated after each event.

This process is defined in the following way:

Definição 1 *Let M be the set of all the agent models; let A be the set of all possible actions; let E be the set of all events. Let $a_1, ..., a_n$ be actions and let $e_1, ..., e_n$ be events such that $a_i \in A$, $e_i \in E, 1 \leq i \leq n$, and $act(e_i, a_i)$, $happens(e_i, t, t'), 1 \leq i \leq n$.*

Let m be an agent model such that $m = < RR, BR, Ac, T, WK > \in M$, where RR are the reasoning rules, BR the behavior rules, Ac the actions description, T the temporal axioms, and WK the world knowledge.

The update function is defined as: $update : M \times E^n \longrightarrow M$, such that:

1. $$update(m, e_1 \times ... \times e_n) = < RR_1, BR_1, Ac_1, T_1, WK_1 >$$

2. $RR_1 = RR$, $BR_1 = BR$, $Ac_1 = Ac$, $T_1 = T$, *and*

3. $$WK_1 = WK \cup \left\{ \begin{array}{c} act(e_1, a_1), happens(e_1, t, t'), \\ ..., \\ act(e_n,a_n), happens(e_n, t, t') \end{array} \right\}$$

the world knowledge is updated with the new events.

The new agent model is the well founded model of the new extended logic program.

However, this update process may create inconsistency, initiating properties that are contradictory with existing ones. In these situations it is necessary to revise the updated program, terminating properties that support the contradiction. In fact, contradiction is created by the violation of integrity rules of the form:

$$\leftarrow holds\_at(P_1, T), holds\_at(P_2, T).$$

In order to remove the contradiction we have used the Contradiction Removal Semantics from the work of Pereira et al. ([Dam96; SDP96]).

5 Examples

In this section we will present some examples that show how the degree of credulousness and cooperativeness influences the dialogue process.

The following sentence, adapted from Pollack [Pol86], may have different effects depending on the model of the hearer:

- a: I want to talk to Kathy.

This sentence is recognized as the following facts:

$$happens(e_0, t_0, t_1).$$
$$act(e_0, inform(a, b, int(a, talk(a, kathy)))).$$

Suppose the hearer (agent b) is a first degree credulous agent, then (using rule 17):

$$holds\_at(bel(b, bel(a, int(a, talk(a, kathy)))), t_1).$$

Using rule 16 we have:

$$holds\_at(bel(b, int(a, talk(a, kathy))), t_1). \tag{23}$$

Assuming the hearer is a first degree cooperative agent, then (using rule 19):

$$holds\_at(int(b, talk(a, kathy)), t_1).$$

If, on the other hand, the agent is a third degree cooperative agent (he accepts intentions only if they are possible), then he tries to find a sequence of actions that enables the execution of $talk(a, kathy)$. If the sequence of actions does not exists, then the agent does not accept the intention:

$$not\ holds\_at(int(b, talk(a, kathy)), t_1)$$

If the sequence exists, then he accepts it.

Another possible variation can be made if we assume the agent b is a second degree credulous agent (believe only if it is plausible). In this situation, rule 18 would be used and proposition 23 is valid only if it's plausible from b's point of view that the other agent wants to talk to Kathy.

As this example shows, the attitudes supported by an agent model depend directly on the behavior rules of the agent. Moreover, the inferred attitudes will be the input of the agent's planning process and they will define his future actions.

6 Conclusions

We have proposed an agent modeling process with the capability to handle dialogues with agents with different degrees of credulousness and cooperativeness.

It has the following main characteristics:

1. It was defined over a logic programming framework with a specific semantic (well founded semantics of extended logic programs);
2. It has a complete and sound top-down proof procedure;
3. It allows the definition of reasoning and behavior rules. These rules allow the modeling of different behaviors;
4. It has an update and revise procedure defined for any event that may occur;
5. It may be the base of a planning process that allows the participation of agents in dialogues.

However, there are many open problems to be dealt with as future work. First, and as it was pointed out in the previous section, we have not analyzed the integration of the modeling process with the planning process and the natural language generation phase. Moreover, we did not discuss the problem of the recognition of speech acts from natural language sentences. These tasks are pre-conditions for the construction of a robust natural language processing system.

As future work we also intend to integrate this agent modeling framework in a more general architecture allowing a complete representation of dialogues.

References

[ADP95] José Júlio Alferes, Carlos Damásio, and Luís Moniz Pereira. A logic programming system for nonmonotonic reasoning. *Journal of Automated Reasoning*, 14:93–147, 1995.

[Alf93] José Júlio Alferes. *Semantics of Logic Programs with Explicit Negation.* PhD thesis, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, 1993.

[AP96] José Júlio Alferes and Luís Moniz Pereira. *Reasoning with Logic Programming*, volume 1111 of *Lecture Notes in Artificial Intelligence.* Springer, 1996.

[Bra90] Michael Bratman. *What is Intention?, in Intentions in Communication.* MIT, 1990.

[CL90a] P. Cohen and H. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(3), 1990.

[CL90b] Philip Cohen and Hector Levesque. *Persistence, Intention, and Commitment, in Intentions in Communication*, pages 33–70. MIT, 1990.

[Dam96] Carlos Damásio. *Paraconsistent Extended Logic Programming with Constraints.* PhD thesis, New University of Lisbon, 1996.

[DNP94] Carlos Damásio, Wolfgang Nejdl, and Luís Moniz Pereira. Revise: An extended logic progamming system for revising knowledge bases. In Morgan Kaufmann, editor, *KR'94*, 1994.

[Esh88] Kave Eshghi. Abductive planning with event calculus. In *Proceedings of the International Conference on Logic Programming*, 1988.

[Mis91] Lode Missiaen. *Localized Abductive Planning with the Event Calculus.* PhD thesis, Univ. Leuven, 1991.

[Per90] Raymond Perrault. *An Application of Default Logic to Speech Act Theory, in Intentions in Communication*, chapter 9, pages 161–186. MIT, 1990.

[Pol86] Martha E. Pollack. *Inferring Domain Plans in Question-Answering.* PhD thesis, Dep. of Computer and Information Science, University of Pennsylvania, 1986.

[SDP96] Michael Schroeder, Carlos Damásio, and Luís Moniz Pereira. Revise report: An architecture for a diagnosis agent. *xxx*, 1996.

[Sha89] Murray P. Shanahan. Prediction is deduction but explanation is abduction. In *Proceedings of the IJCAI*, 1989.