AgentSpeak(L)

Miriam Kölle

University of Koblenz-Landau
miriamkoelle@uni-koblenz.de

1 Introduction

AgentSpeak(L) is a programming language for agents based upon the BDI model. It was originally developed by Rao in 1996 and this report is basically summarizing his paper[RAO96].
The AgentSpeak(L) language consists of the following main language constructs:

- Beliefs
- Goals
- Plans

While the main components of an agent are:

- Belief Base
- Plan Library
- Set of Events
- Set of Intentions

The belief base consists of all beliefs the agent has about itself and its environment (including other agents). The plan library consists of all plans this agent has access to. Plans are coded by the programmer in advance; the agent is not capable of generating plans itself. All new events are pushed onto the set of events, all plans chosen to be executed are pushed onto the set of intentions. All keywords will be clarified in the following sections.

2 Motivation

Settled in its environment, each agent, perceiving changes of the environment or other agents, should respond to new information in an appropriate fashion. Besides external events, also internal changes might require a reaction. AgentSpeak(L) allows the programmer to specify how the agent should react to events and how the agent can achieve some goal. These instructions are encoded into several plans which the agent can access.
The underlying BDI concept is modelled as follows: all information the agent has about itself, the environment, and other agents is called its *belief* state. As the agent tries to handle an event, it looks up its plan library to find a matching plan for the event. One can call all matching plans, not mandatory applicable in the current situation, the options or *desires* of the agent. Chosing one applicable plan for execution means this process is adopted as an *intention*.

3 Language Constructs

In the following, a brief overview of the AgentSpeak(L) language is provided, basically summarizing Rao's work [RAO96].

3.1 Beliefs

Beliefs represent information that the agent has about its environment. AgentSpeak(L) models beliefs as predicates, representing the corresponding property or relationship.
Example: at(X) (the agent believes being at position X.)

3.2 Goals

Goals are states that an agent wants to effect. AgentSpeak(L) distinguishes between achievement goals and test goals: an achievement goal denotes a state the agent wants to effect where the atom is a true belief. An achievement goal is denoted with an exclamation mark placed in front.
Example: !at(x) (the agent 'wants' to be at position x, thus at(x) shall be a true belief.)
Test goals offer the opportunity for an agent to test whether the atom is a true belief or not. Test goals are annotated with a preposed question mark.

3.3 Events

The agent needs to react to changes. These can be external or internal changes, i.e., changes that concern the environment and that demand for an update of the agent's belief base, or changes that concern the 'inner' of the agent, that is changes in its goals as a consequence of additive subgoals while pursuing another goal or due to agent communication. In both cases the agent needs to handle these events.
According to the previous section do exist six different events as consequences to changes, namely addition or deletion of beliefs, achievement goals, and test goals:

- $+b(t)$ addition of a belief
- $-b(t)$ deletion of a belief
- $+!g(t)$ addition of an achievement goal
- $-!g(t)$ deletion of an achievement goal
- $+?g(t)$ addition of a test goal
- $-?g(t)$ deletion of a test goal

In order to handle a new event in a proper fashion, the agent has to have a plan telling it what to do as a reaction to the event. For this reason, each agent has a plan library, consisting of plans that define reactions to specific occuring events.

4 Plans

A plan is an event-invoked 'recipe' specifying how an agent shall react to an event and how it shall behave in order to achieve goals. A plan consists of a head, including the triggering event and the context, and a body. The syntax of a plan is given by
$TriggeringEvent : context < -body.$
The three components settle the following:

- The triggering event is the event that this plan can handle.
- The context consists of a sequence of beliefs, that must hold for the plan to be applicable. That is, the agent has to believe that the context (i.e., the beliefs stated there) is true.
- The body contains a sequence of actions or (sub)goals.

Example 1 (taken from):

$$\begin{aligned} &+need(It) : \\ &price(It, P)\ \&\ bank\_balance(B)\ \&\ B > P \\ &< - !buy(It). \end{aligned}$$

Example 2 (taken from [BH05]):

$$\begin{aligned} &+!location(X) \\ &: notlocation(X)\ \&\ safe_path(X) \\ &< - move\_towards(X); \\ &!location(X). \end{aligned}$$

All plans are included in the agent's *plan library*. All plans triggered by an specific event are called *relevant plans*, i.e., their triggering event matches. All relevant plans with a true context are called *applicable plans*, i.e., they are executable.

5 Control Loop

In sum, the control loop of an AgentSpeak(L) agent is as follows:

- Perceive the environment
- Update belief base
- Select an event
- Retrieve all relevant plans
- Retrieve all applicable plans
- Select one applicable plan
- Select an intention
- Execute first step of an intention

In general, several events might occur in one simulation step, several plans might be applicable and several intentions executable. Therefore, select functions are required (we can customize these using Jason).
All events are added to the set of events. A select function selects one event and this event is removed from the set of events. Now all relevant plans are searched, followed by all applicable plans out of the relevant ones. A select function selects one of these applicable plans. It is added to the set of intentions.
(Note: every intention is a stack and the set of intentions consists of all these stacks. An external event causes a new stack, an internal event effects that the intended plan is pushed onto the existing intention that triggered this internal event.) A select function selects one intention to execute (goal or action).
Executing an achievement goal means: generate an internal event, chose applicable plan, and push this plan on top of the current intention. Executing an action means: add this action to the set of actions.
Summarized, an agent can be modelled as a tuple $< E, B, P, I, A, S_E, S_O, S_I >$ where E is a set of events, B a set of beliefs, P a set of plans, I a set of intentions, A a set of actions and S the three select functions to select an event, applicable plan, and intention.

References

[BH05] Rafael H. Bordini and Jomi F. Hübner. 2005. BDI agent programming in agentspeak using Jason. In Proceedings of the 6th international conference on Computational Logic in Multi-Agent Systems (CLIMA'05), Francesca Toni and Paolo Torroni (Eds.). Springer-Verlag, Berlin, Heidelberg, 143-164.

[RAO96] Anand S. Rao. 1996. AgentSpeak(L): BDI agents speak out in a logical computable language. In Proceedings of the 7th European workshop on Modelling autonomous agents in a multi-agent world : agents breaking away: agents breaking away (MAAMAW '96), Walter Van de Velde and John W. Perram (Eds.). Springer-Verlag New York, Inc., Secaucus, NJ, USA, 42-55.