Actions with Durations and Failures in BDI Languages

Louise A. Dennis and Michael Fisher

Abstract. BDI (Belief-Desire-Intention; [3]) languages typically model interaction with the environment either as an action or as a capability. It is generally implicitly assumed that these do not take long and usually execution of the program waits for them to complete before processing other intentions, goals and plans.

Agents are increasingly being used as the discrete components of hybrid systems. In these situations real actions may actually take considerable time to complete. Therefore, we do not want the agent program to suspend, but to continue operating, in order to perform error monitoring etc. Although ad hoc solutions to this problem exist, these frequently involve treating the action/capability as the initation of an interaction with the environment. Perception is then used to judge when the interaction has concluded. Furthermore an action may need to be aborted while it is still executing.

Several life-cycles for goal processing in BDI agents have been proposed. It is agreed that goals need to transition through a number of states, including a Suspend state in which execution of any plans associated with the goal is halted and an Active state in which the goal is being processed. We adopt the semantics presented by Harland et al. [2] This semantics provides a comprehensive account of the goal life-cycle presented in the generic CAN (Conceptual Agent Notation) formal system [4] which we also adopt here.

Goals are represented as a tuple, $(I, G, Rules, State, P)$ where $I$ is a unique identifier for the goal, $G$ is the goal type (a (achieve), m (maintain), v (perform) or t (test)), Rules are a set of condition-goal action pairs. These govern how the goal moves between states. State is the current goal state and $P$ is the current plan body associated with the goal (if any). Further, $\epsilon$ indicates the absence of any plan body, nil a trivially successful plan and fail a trivially unsuccessful plan. Harland et al. [2] assume that means-end reasoning is employed to select plan bodies. In our semantics for the addition of capabilities to this framework we will sometimes choose to specify the outcome of this means-end reasoning.

This paper is a shortened version of [1].

1 Introduction

BDI (Belief-Desire-Intention; [3]) languages typically model interaction with the environment either as an action or as a capability. It is generally implicitly assumed that these do not take long and usually execution of the program waits for them to complete before processing other intentions, goals and plans.

Agents are increasingly being used as the discrete components of hybrid systems. In these situations real actions may actually take considerable time to complete. Therefore, we do not want the agent program to suspend, but to continue operating, in order to perform error monitoring etc. Although ad hoc solutions to this problem exist, these frequently involve treating the action/capability as the initialization of an interaction with the environment. Perception is then used to judge when the interaction has concluded. Furthermore an action may need to be aborted while it is still executing.

Several life-cycles for goal processing in BDI agents have been proposed. It is agreed that goals need to transition through a number of states, including a Suspend state in which execution of any plans associated with the goal is halted and an Active state in which the goal is being processed. We adopt the semantics presented by Harland et al. [2]. This semantics provides a comprehensive account of the goal life-cycle presented in the generic CAN (Conceptual Agent Notation) formal system [4] which we also adopt here.

Goals are represented as a tuple, $(I, G, Rules, State, P)$ where $I$ is a unique identifier for the goal, $G$ is the goal type (a (achieve), m (maintain), v (perform) or t (test)), Rules are a set of condition-goal action pairs. These govern how the goal moves between states. State is the current goal state and $P$ is the current plan body associated with the goal (if any). Further, $\epsilon$ indicates the absence of any plan body, nil a trivially successful plan and fail a trivially unsuccessful plan. Harland et al. [2] assume that means-end reasoning is employed to select plan bodies. In our semantics for the addition of capabilities to this framework we will sometimes choose to specify the outcome of this means-end reasoning.

This paper is a shortened version of [1].
This precondition determines the location of the robot and the direction it is facing. We assume perception informs the agent when its motors are engaged (motors_on). It has no abort condition.

The agent’s goal is to reach (0, 2) identified as, r02. The agent’s initial belief base is \{at(0, 1), angle(0)\}. As a result of means-end reasoning it adopts the plan move(1). The agent state at this point is shown in (1). For presentational reasons we have split the tuple \{pp, cp, angle, motors_on\} and only show the single goal we are interested in, not the set of all goals. After (9) fires, the motors are engaged, and the goal is suspended. The new state is shown in (2). Beliefs continue to be updated via perception. Eventually the agent believes at(0, 2) and no longer believes motors_on. (10) fires and the agent is in state (3).

If the robot at (0.25, 0.3) when the motors stop, the agent transitions using (12) to (4).

We add two beliefs, cp(X, Y) and pp(X, Y) – the agent’s current and previous position – and an abort condition and capability:

\[
\langle move(D), at(X, Y) \land angle(0), \top, \neg motors_on \land at(X + D\sin(0), Y + D\cos(0)), \neg motors_on \land \neg at(X + D\sin(0), Y + D\cos(0)), motors_on \land pp(X', Y') \land cp(X'', Y'') \land \\
\sqrt{(X + D\sin(0) - X')^2 + (Y + D\sin(0) - Y')^2 >} \\
\sqrt{(X + D\sin(0) - X'')^2 + (Y + D\sin(0) - Y'')^2} \\
\langle abort(move(D)), motors_on, \top, \neg motors_on, \perp, \perp \rangle
\]

If the agent’s state is that shown in (5), it transitions, via (11), to (6). The abort capability is invoked and (9) transitions the agent to (7). Assuming the motors stop this will then transition to (8).

4 Conclusion

We have argued that BDI representations of interactions with the environment need to account for actions taking time to complete and aborts. We have extended the semantics for the life-cycle of goals presented in [2] to show how the declarative representation of capabilities with durations, failures and aborts can be integrated with this semantics.

REFERENCES