

Embedded Agents

An agent that operates in a real environment (robot, Internet softbot) faces many difficult problems. Three of the most important are:

- the agent's planning must be interleaved with its acting — need *incremental execution*;
- the agent only has *incomplete knowledge* and must *sense* the environment to decide what to do;
- the agent operates in a *dynamic* environment — there are other agents who act and the agent must detect this through sensing and update its beliefs accordingly.

Embedded Agents and IndiGolog

Yves Lespérance

July 1999

Incremental Execution

Nondeterministic search is how Golog and ConGolog support planning — a key ingredient of intelligence.

But search, or planning, or exploring your options, is something you do in your head before you act. At some point, you must stop thinking and start acting.

An agent that has a fairly simple task to accomplish can do all its planning first, and then execute the plan it came up with, and then be done.

Golog and ConGolog were conceived according to this simple model: search all the way to a final situation of your (nondeterministic program) and return the situation; then you can execute it if you want.

Incremental Execution (cont.)

But an agent that has a complex task to accomplish and must run for a long time cannot do all his planning before it acts; it must do some planning, then execute some of the plan it constructed, then do some more planning, then more acting, . . .

For this, the simple Golog/ConGolog execution model is inadequate. We need a version of the language where *search is interleaved with execution*.

Incomplete Knowledge and Sensing

Another problem: agents have *incomplete knowledge* and must perform *sensing actions*.

E.g. an agent who must go to the airport and board a flight; it can't know which flight gate to go to in advance; it must do sensing once it is at the airport to find out!

Golog/ConGolog doesn't support this. (The account of knowledge seen last week support modeling dynamics of knowledge, but the notion of successful terminating execution in Golog must be changed to take this into account.) See Levesque's paper "What is planning in the presence of sensing?".

In general, the agent needs to generate plans that includes sensing actions and branching. For this, it needs to search over the possible outcomes of sensing actions. Bad combinatorics!

Executing plans that contain sensing actions is simpler. In some cases, it's sufficient.

Operating in a Dynamic World

A third problem: the world is dynamic. Other agents perform actions. Even if agent has complete knowledge initially, it doesn't stay that way.

In some cases, the agent can easily determine what exogenous actions have occurred (through sensing). Then, we can have the executor monitor for these and incorporate them into the execution.

In general, the agent need to diagnose what exogenous actions have occurred to explain sensing data. This is similar to a planning task. Hard!

IndiGolog

IndiGolog addresses all 3 of these problems. It supports:

- interleaving search and execution; arbitrary choice of next action and execute; but adds search block construct;
- incorporating exogenous actions into the execution (user must define monitoring routines);
- execution of sensing actions and incorporation of sensing data into execution history; this data can be used by later tests; uses a dynamic closed world assumption.