

AM-PSI

Advanced Magic GmbH

28. Oktober 2010

Advanced Magic GmbH
Hauptstr. 9
96120 Bischberg
Germany
www.advancedmagic.de
+49/951/9685091

Contents

1	Introduction	3
2	Objectives	4
3	Implementation	5
4	Operation	6
5	To Do	7

1 Introduction

AMPsi is an autonomous software agent. It was inspired by Prof. Dr. Dietrich Dörner's original PSI program written in Turbo Pascal for the simulation of the human mind.

2 Objectives

AMPsi is designed from scratch to implement many of Prof. Dr. Dieter Dörner's findings about how the human mind performs. The idea is, to build a reference implementation of a universal autonomous problem solving software agent.

3 Implementation

AMPsi is written in Python, a widely used, very elegant interpretive language. Python is perfect for explorative programming, both easy to write and easy to read.

AMPsi can simulate an arbitrary number of agents within one world.

The implementation consists of two programs: `welt.py` and `ampsi.py`. The first one, “`welt.py`”, (“Welt” is German for “world”) displays the virtual world, the environment, the agent(s) live(s) in. The `welt.py` communicates with the agents via a shared text file called “`welt`”. The world simulator runs as a single process as does each agent process. The agents modify the shared file and the `welt.py` process displays it.

The world of AMPsi consists of an array of upper case letters (A through Z) and spaces (blanks) arranged as 20 lines of 40 characters. The letters may signify food (“E”) or drink (“W”) or nothing special. Food and drink improve the wellbeing of the agent: they satisfy its needs. Some other letters induce negative feelings (i. e. pain) in the agent.

The agent is capable of moving in its world with a limited number of operators. The operators “`move_up`”, “`move_down`”, “`move_left`” and “`move_right`” move the agent in the respective direction one cell at a time. The agent has two operators to pick up food and drink called “`eat`” and “`drink`” of course.

During its lifetime the agent needs both food and drink in order to feel well. When levels of hunger and thirst have passed certain thresholds, the agent tries to satisfy these needs.

When the agent is first created, it doesn’t know, how to satisfy its needs or how to avoid pain. It learns by trial and error. It needs to explore the world. Since it doesn’t know how to use its operators, all it can do is try them at random.

Eventually the agent has learned how to satisfy its needs, how to satisfy hunger and thirst and avoid pain: the agent replays automatisms to perform task important to its survival.

In order to avoid pain or find other satisfying situations, the agent must be able to recognize situations it has visited before. However, a situation may not look exactly like it looked before. Moreover the agent should be able to recognize certain aspects of a situation relevant to its wellbeing. This means, the agent should be able to avoid dangerous (i. e. painful) situations even if it has never seen the particular new situation. It should be able to recognize food and drink in new situations too. The agent must be able to generalize and abstract situations and apply this knowledge to new situations. This is solved in view’s “`similar`” and “`match`” functions.

4 Operation

Start the simulation by hitting function key F2 or click on menu Simulation->Start. Stop the simulation by hitting function key F3 or click on menu Simulation->Stop. To run in single step hit F4 or click on Simulation->Single Step.

To retrieve information about nodes and situations click on menu Dump and the appropriate submenu. To use the Dump->Show Path feature, open the protocol window before. Then select the protocol nodes in the list box you want to know the path of. After that, you find the path graphically drawn in the protocol window.

Update information in either of the Dump windows by clicking on the appropriate menu item again, or hit the corresponding function key again.

To view detailed information on the graph drawn, click in the graph with the mouse. The detail window appears and shows more information. Click again in the graph to update the detail window.

5 To Do

- Implementation of most human motives including competence, certainty, exploration, affiliation.
- Measurement of psychological internal states for competence and certainty.