

Programming with Python
Part 4
First Demo complete
emerging-mind.org eJournal ISSN 2567-6466
(info@emerging-mind.org)

Gerd Doeben-Henisch
gerd@doeben-henisch.de

October 20, 2017

Contents

1	First Milestone Reached	1
2	How to Continue	2

Abstract

Part 4 is the first Milestone: a complete GUI-based application in python with a simple environment (20 percent objects, randomly distributed, one food object) and exactly one actor moving around. In the beginning you can choose the number of cycles which should be run and the type of behavior function, you want to test. Actually there are three options: manually (:= 0), fixed (:= 1) as well as random (:= 2). With option 1 or 2 the demo is running by its own. With option 0 you have to select the next direction manually. The application stops either when the number of maximum cycles has been reached or when the actor died on account of energy exhaustion.

1 First Milestone Reached

The basic idea from the beginning was to check whether it is possible to program in python a simple actor-environment demo with a graphical user interface (GUI).

During the parts 1-3 step wise it could be proved that all the wanted features could be realized.

Clearly the actual code is far from being elegant nor is he completely explained. But everybody can inspect the code delivered as a ZIP-folder.

This first GUI-based demo contains the following features:

1. A 7 x 7 grid is offered with empty cells.
2. 20% are randomly occupied by obstacles (black cells).
3. 1% of cells is randomly occupied by food (i.e. here one cell) (green cell).
4. The start position of an actor (red circle) can be selected by pointing into the grid.
5. The number of cycles can be entered how often the environment shall be repeat the event loop.
6. The type of behavior function of the actor can be selected: manually (:= 0), fixed (:= 1) as well as random (:= 2). With option 1 or 2 the demo is running by its own. With option 0 you have to select the next direction manually by clicking into the grid.
7. While the demo is running it reports the actual energy level of the actor as well as the actual 'Time of the environment' (which corresponds closely to the number of cycles).
8. If either the maximum number of cycles is reached or the energy of the actor is below 0 the application will stop and after 10 s vanish from the screen.

2 How to Continue

There are many options how to continue. Actually the following ones are considered:

ACTOR-ENVIRONMENT FEATURES:

1. Enhance the actual version with with e.g. offering the selection of more parameters to be eligible.
2. Allow multiple actors simultaneously.

3. Allow the automatic repetition of a whole experiment over n-times.
4. Allow storing of results and statistical evaluations.
5. Start explorations of different behavior functions like genetic algorithms, classifiers, reinforcement, learning similar to alpha go zero, etc.
6. Enhance perceptual structures and motor patterns.
7. Check different cognitive architectures
8. Enable basic dialogues with the environment
9. ...

COMPUTING FEATURES:

1. Transfer the windows implementation on ubuntu 14.04 too.
2. Compare the different versions.
3. Integrate the actor-environment pattern within the ros-architecture
4. Allow real-world sensors and actors especially for robotics, for sound art work, for picture art work, for sculpture art works as well as for text art work)
5. Rewrite the actor-environment demo as a distributed process network.
6. Realize a first demo of a *Local Learning Environment*
7. ...