Department of Computer Science, Electrical and Space Engineering

D7041 Applied Artificial Intelligence

Lab work: Intelligence through emergent behavior and Interaction in Multi-Agent system

Part 1. Mars Explorer System (Emergent behavior)

In this part, you need to develop a system that collects precious stones on Mars. The system can be developed as a complex machine with intelligent algorithms and big software.

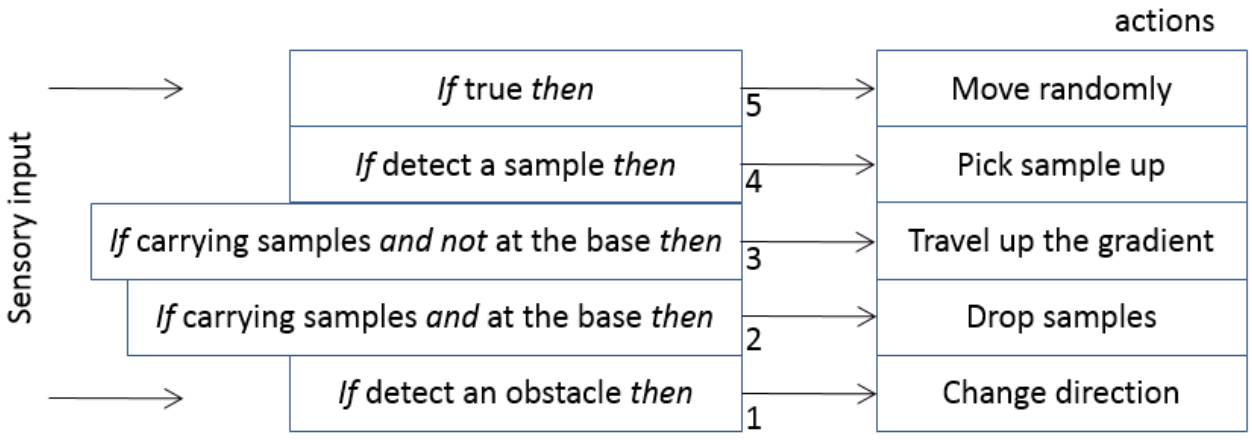
Instead, your task is to develop intelligent system though emergent behavior. You will develop a simple agent that implements a few functions (instructions). A swarm of these agents will constitute the system. Agents like ants will find the rocks and bring them to the Mother ship.

Goal: Get familiar with programming through distributed agent-oriented perspective. Have experience developing emergent behavior.

Agent's program is described below. The architecture is called *Subsumption* architecture. These agents are reactive. More on this is in the lecture 1.

You can design the "World" as you want: mothership can be in the center or the corners. The Mothership is stationary, its coordinates are known to the agents.

Best working solution we can send to Northwestern University to add to default models distributed with the NetLogo installation.



Figur 1. Agent reactive architecture and its behaviors.

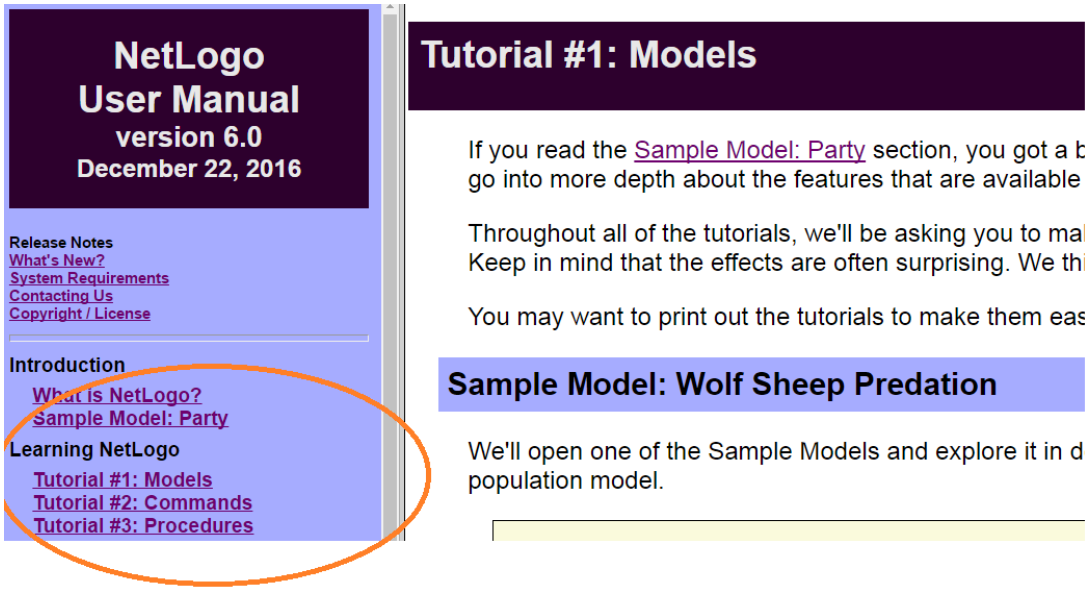
We will use NetLogo software to develop this system. It is easy to learn and has visualization inbuilt. The software was developed by Wilensky, Northwestern University's Center for Connected Learning (CCL) and Computer-Based Modeling, Northwestern University, Evanston, IL , 1999.

NetLogo can be downloaded from https://ccl.northwestern.edu/netlogo/

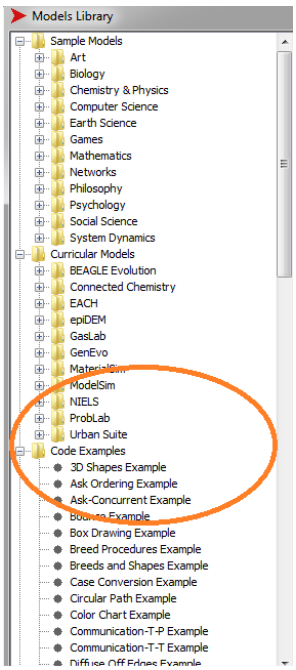
It is also on LTU Canvas site for this course.

Tutorial on NetLogo

In order to get started, you need to complete three tutorials https://ccl.northwestern.edu/netlogo/docs/



The examples of code in NetLogo can be found here.



1. Quick guide http://luis.izqui.org/resources/NetLogo-6-0-QuickGuide.pdf
2. NetLogo: File->Models Library -> Code examples (see picture on the right)
    a. Example maybe useful: "Random walk", "Random Grid Walk", "Hill climbing", "Scatter", "Patch coordinates" and more.
3. NetLogo: File -> Models Library -> Biology->Moths
4. NetLogo: File -> Models Library -> Biology->Termites
5. NetLogo: File -> Models Library -> Biology-> Ants
6. NetLogo: File -> Models Library -> Biology-> Ant Lines
7. NetLogo: File -> Models Library -> Biology-> BeeSmart Hive Finding
8. https://ccl.northwestern.edu/netlogo/models/community/index.cgi

If you need support using NetLogo try this

1. https://groups.yahoo.com/neo/groups/netlogo-users/info
2. https://ccl.northwestern.edu/netlogo/docs/programming.html
3. https://ccl.northwestern.edu/netlogo/docs/dictionary.html
4. https://ccl.northwestern.edu/netlogo/docs/interface.html
5. https://subversion.american.edu/aisaac/notes/netlogo-intro.xhtml

Part 2. Game theory: Hawk/Dove

In the second part of the lab, you need to develop Hawk Dove game.

Again, it is up to you how you want to design the "World".

The interface should allow user to enter payoffs for each strategy against another and initial population size of each strategy players.

The interface should show current population (proportion).

User should clearly see the result of the game. (You can select different colors for the turtles that play different strategy).

Game explanation and payoff matrix is described in the lecture.

Goal: Get familiar with the idea of strategy and payoffs. Play around with the payoff matrix, and see effect on the result and proportion of the population. Find stable strategies.

Tool is the same – NetLogo.

Might be useful: NetLogo - > Menu - > Models library -> Sample Models ->

1. Biology-> "Wolf Sheep Predation"
2. Social Science -> Voting
3. Unverified -> Prisoner's Dilemma
4. Computer Science -> Cellular Automata -> Life