Agents and Multi-Agent Systems (6CCS3AMS/7CCS3AMS)
Coursework 1: Forest Fires and Reactive Agents[1]

Domain description

One of the tasks of the U.S. Forest Service is to detect and control fires in their national parks. The task involves constant monitoring of the forest area. An immediate response is required should a fire be detected, so that the spread of the fire is minimised. Preliminary research of past cases of wildfires has demonstrated that if detection of a fire is immediately followed by an attempt to suppress the fire, even with limited means, then in many cases the wildfire can be prevented.

Given that constant monitoring of the forest area is a hard task, and that even when a fire is detected a certain amount of time is needed for the ground units to arrive at the location of the fire, the U.S. Forest Service is investigating the development of autonomous ground fire-fighting units that will constantly patrol national parks and act appropriately, i.e. an agent-based system. Such a system consists of a number of agents that are responsible for detecting and extinguishing fire. One approach to such a system is to model each ground unit as a reactive agent; the overall behaviour of each agent is the result of individual behaviours organised in a *subsumption architecture*, as proposed by Brooks. Each behaviour is expressed as a rule that defines the course of action given the current state of the world, which is perceived by the agent's set of sensors. (Note that **reactive agents and the subsumption architecture are covered in the lecture on Monday 15 October**.)

In such a detection/suppression system we are concerned with two factors:

1. The number of dead trees, i.e. trees that the agents could not extinguish on time.
2. The number of saved trees, i.e. trees that were on fire but the agents managed to extinguish.

Of course, an ideal system would extinguish fires with the minimum number of dead trees and this is the aim of the final system.

The NetLogo model

A simulation of the above domain has been implemented as a model in the NetLogo platform. The code of the model is available for you to download from the KEATS page (named *cw1-code.nlogo*). **Note, this model will not run without modification on the most recent version of NetLogo; please make sure you are using version 5.3.1.**

The model contains:

1. The code for simulating ground units, i.e. facilities for creating the initial population of such units (`fire-units-num`), a number of sensors (`detect-obstacle, detect-fire` etc.), a number of effectors (`put-out-fire, turn-randomly, move-ahead` etc.), and a preliminary implementation of the agent's behaviour following the subsumption architecture. Currently the design of the agent contains only the following two rules, which allow the agent to move around and avoid obstacles (i.e. other units and trees on fire):
   - if `detect-obstacle` then `avoid-obstacle`
   - if `true` then `move-randomly`
2. The code for creating a base and setting up a radio signal that can be used to guide agents towards the base.
3. The code for simulating fire events in the forest. The simulation environment includes creation of the initial forest environment with a varying number of trees (`tree-num`), a mechanism for randomly creating a number of initial fire spots (`number-of-fires`), and a

[1] With kind permission from the authors, adapted from the coursework described in:
Ilias Sakellariou, Petros Kefalas and Ioanna Stamatopoulou, "Teaching Intelligent Agents using NetLogo", in *Proceedings of the ACM-IFIP Informatics Education Europe III Conference*, 2008.

simple behaviour model for fire spreading in the forest area. In this simulation, each tree that is on fire spreads the fire to its neighbouring trees (if any) and burns for a number of execution cycles. If the fire is not extinguished after a number of cycles, the tree dies.

4. Facilities for monitoring various parameters of the performed experiment such as the number of dead trees, the number of saved trees, etc.

In the NetLogo model, ground units are represented as `turtles` of breed `units`, living trees as `turtles` of breed `trees`, trees on fire as `turtles` of breed `fires`, and extinguished trees as `turtles` of breed `fires-out`. Fire spots are created randomly and with a probability on each execute cycle, so that fire events occur at different time points. The simulation stops automatically when all fire spots have been created and all trees on fire either have died or have been extinguished by the ground units.

Tasks

1. As it stands the model is not very useful since ground units do not put out any fires. Using the set of sensors and effectors provided (i.e., do not add any new sensors or effectors) extend the code that determines an agent's behaviour (i.e., modify the `execute-behaviour` procedure) so that it is able to effectively extinguish fires (**the extended agent should still use the subsumption architecture**). In your design you should consider that each ground unit can carry a limited amount of water, which is consumed as units extinguish fires in the forest. One unit of water is required to extinguish the fire on one tree. If the water supplies of the unit are exhausted, then the unit must return to the station to reload. You should provide:

   - A brief description and justification of the agent design you propose for the ground units (no more than 300 words). **You must include *documented* code snippets of *all* new and altered code to help to describe your agent.** Do not include the whole NetLogo model (the code is not included in the word count).
     (2 marks)
   - A *.nlogo* file of your complete *documented* code for your agent system.
     (2 marks)

2. An experimental evaluation that shows how the performance of your extended model developed for task 1 varies for different starting conditions. Note that there are four different parameters you can modify. For each experiment you run, you must fix `tree-num` as 400, and you must fix `number-of-fires` as 40. You should vary only `fire-units-num` and `initial-water` to explore the performance of your model. (Note that the speed of ground units depends on the amount of water they are carrying; units that are loaded with a large amount of water move more slowly than units loaded with less water.) Please provide:

   - A clear description of the set of experiments used to evaluate the performance of your extended model. This should include details of the parameters investigated, why you selected those parameters, and what you intend to demonstrate from the experiments. (No more than 400 words.)
     (3 marks)
   - A set of results obtained from your experiments detailed above (your results can be presented in graph or table format) together with an analysis of the results. Include details of any conclusions you have drawn from your analysis, and any justifications of your conclusions. Briefly comment on any limitations of your results. (Note, it is not necessary to provide any kind of statistical analysis.) (No more than 400 words.)
     (3 marks)

3. Propose improvements to the above agent system that could increase its efficiency. Note that the current limitations on the sensors and effectors are fixed (for example, you cannot just propose an increase in the range of the sensors provided) but you may introduce new types of sensors or effectors if you wish, or new types of agent. The agents must still be reactive and use a subsumption architecture. Note that you are not expected to implement the code for your proposal, or to discuss implementation specific issues related to NetLogo — but the description of your proposed improvements should be detailed, and should include a justification of why you think your proposed improvements will increase the efficiency of the agent system (no more than 400 words). **(5 marks)**

4. Discuss the advantages and disadvantages of the reactive agent approach to the forest fire problem (no more than 250 words). **(3 marks)**

Submission

Please ensure that you upload your submission with the specified formats and that you stick within the word limits. **Failure to do so may lead to you receiving 0 marks for your submission.** You should submit two files:

- A written report (.pdf format) that contains appropriate solutions to each task above. **For each task, please start the answer on a new page.**
- *Documented* NetLogo code (.nlogo format) for your solution to Task 1. Results to your experiments (detailed in your report) should reflect those obtained by running your submitted code.

Assessment criteria

Note that the focus of the coursework is to assess your understanding of reactive agent architectures, and your ability to evaluate the performance of such systems based on simulation results. Your submission will be assessed according to the following criteria:

- Correctness, originality, and justification of the proposed solutions
- Analysis and presentation of experimental results
- Implementation and code documentation
- Presentation of the report (clarity and structure)