UNIVERSITY OF SOUTHERN DENMARK

AM 24 - MULTI-AGENT SYSTEMS
EXAMINATION

ODENSE, AUGUST 2010

On a 2-dimensional grid G representing the surface of a planet, **N** bases land randomly to collect ore samples. The planet is a torus. The ore is distributed on the planet with a density **D**. Coordination mode **M** of the bases is either cooperative when the bases are belonging to the same company, or competitive when they involve different companies. Each base carries societies of robots consisting in **X** explorers and **Y** transporters. Robots can move on the grid in the eight directions. Two robots cannot be located at the same time at the same place. Each robot has a finite perception scope **P** ($1 < PxP \leq G/100$) and a finite communication scope **I**. This also applies to the bases. Robots can tune their perception scopes by themselves. When an ore sample is in the range of the perception scope of an explorer, the explorer can acquire its coordinates through perception. Explorers can send coordinates of an ore sample it has detected. Robots and the bases have a limited size memory to record a maximum of **S** ($S < X+Y$) coordinates. When a transporter is at the same position of a sample of ore, it can pick the ore sample. Transporter can transport samples back to the base and deposit them here. The capacity **C** of the base is limited. Robots have a limited capacity **W** of ore sample they can grab and/or carry at a given time. At each cycle, each robot executes only one action: a message sending, a perception requesting, a body moving, an ore sample picking, … Robots have limited energy that they consume when acting: a perception request is costing $\sqrt{(P)}$ units when a message sending is costing 1 unit, whilst robot motion is the far most expensive action among all possible actions. A robot simply dies when its energy is consumed in its batteries. A robot can acquire additional energy at the base. When the base is full of ore or after a given time **T**, robots alive return to the base. Delivery of the solution provided by your system should contain the quantity of ore collected, the time spent for it, and the percentage of robots alive.

03-04-05 August

• Install Madkit (see www.madkit.org) and become familiar with it through online documents and available examples. In case you find it appropriate, you are allowed to purposively reuse parts of the code you may find there.

05-06-09 August

• Analyse the domain and the problem of this foraging robots application, and extend specifications if needed. Justify your extensions.
• Perform the analysis and design for the requested MAS. Pure reactive MAS are forbidden, which means that some agent reasoning is expected in your system. Explicit this part in your final report.
• Discuss the A, the E, the I, the O, and the dynamics models you are selecting (cf. lectures until 06 August). In particular:

- For every type of agent you will define, you should report the state graph describing its life cycle.
- You should explain every additional variable you consider as being particularly interesting for each type of agents, what it does represent, and how it should evolve during the execution of the your program.

• Implement your first MAS system with MadKit, and experiment it until it runs adequately. To help in the debug and testing,

- Each measurable quantity (time cycle, ore quantity…) should be associated with an external variable.
- The input user interface of your software should permit to initialize the initial values of : **C** (capacity of each base in number of ore samples), **D** (density of ore, as a percentage), G (size of the grid, **I** (fixed communication scope), **M** (coordination mode between bases, cooperative = 1 or competitive = 0), **N** (number of bases), P (initial perception scope), **S** (memory size of each robot), **T** (maximal number of cycles), **W** (maximal number of ore a robot can grab and/or carry) **X** (number of explorers per base), **Y** (number of transporters per base), and the ones of the additional important variables you have defined.
- The code should contain the necessary flags and comments indicating when agents enter and leave their states, also where and when the communicating primitives are activated.
- The output user interface should generate a file containing the results, including the values of the quantity of ore collected, the time spent for it, and the percentage of robots alive.

06-09-10 August (parameters and experimental conditions to be announced on August 09)

• Experiment the ratio X explorers / Y transporters and study how it affects convergence.
• Experiment several energy functions of the robots and study how they affect convergence.
• Experiment different communication and perception scopes and how they affect convergence.
• Revise your MAS system with MadKit, experiment it until it runs adequately for one base.

09-10-11 August (parameters and experimental conditions to be announced on August 11)

• Experiment the extension of the system when several lunar ships are cooperating to collect the ore.
• Experiment the extension of the system when several lunar ships are competing to collect the ore.
• Experiment several ratios of competing bases and study how this distribution affects the results.
• Revise your MAS system with MadKit, experiment it until it runs adequately for several bases.

10-11-12 August

• Write a 15-20 pages (maximum, including figures) PDF report including analysis and design, models, experiments, results, self-evaluation and proposed extensions. Take care that the agent architectures as well as interaction and organisation structures, more generally that all the choices made, are clearly reported and argued.
• Send your report and software to Y. Demazeau, B. Grønbæk and C. Risager before 12 August 2:00 pm. In your mail, argue a set of recommended values for W, X and Y with (X+Y)*W = 20 for the given setup C = 200, D = 5 %, G = 200x200, I=40, M = 1, N = 1, P = 20, S = 19, T = infinite.
• Prepare a 30 mn slide presentation of your work using slides. Send the handouts to Y. Demazeau, B. Grønbæk and C. Risager 12 August 6:00 pm. You are not allowed to modify your slides after this deadline

13 August

• Deliver your presentation on 13 August between 9:00 am and 4:00 pm. The presentation will be followed by a set of individual questions.
• Check the report and the software of another student group (to be indicated on 13 August 9:00 am) during the half day you are not presenting your own work and report any problem to Y. Demazeau, B. Grønbæk and C. Risager 13 August 6:00 pm.