Hwk 3: Modeling, Distributed Task Allocation, and Sensor/Actuator Networks

This homework requires the following software:

- Webots

Note that the homework is graded.
For any questions, please contact us at **dis-ta@groupes.epfl.ch**.

1.1 Information

In this assignment, you will find several exercises and questions.

- The notation $\mathbf{S}_x$ means that the question can be solved using only additional simulation.
- The notation $\mathbf{Q}_x$ means that the question can be answered theoretically, without any simulation; your answers should be submitted in your report. The length of answers should be approximately **two sentences** unless otherwise noted.
- The notation $\mathbf{I}_x$ means that the problem has to be solved by implementing some code and performing a simulation.
- The notation $\mathbf{B}_x$ means that the question is optional and provides bonus points. It should be answered if you have enough time at your disposal.

1.2 Outline

This homework is mainly dedicated to test your understanding of the 7^{th}, 8^{th} and 9^{th} laboratory.

1.3 Grading

The report for this homework will be due on **Thursday December 6 at 12h00 noon**. **Late submissions will be penalized**, starting two hours after the deadline, we will take 5 points from your homework result per hour past since the deadline. Please format your solution as a **PDF file** with the name **[name]_hwk[#].pdf**, where **[name]** is your account user name and **[#]** is the number of the homework assignment, and upload it as the Report submission onto Moodle. If you have additional material (movies, code, and any other relevant files), upload a **[name]_hwk[#].zip** archive or a **[name]_hwk[#].tar.gz** archive instead with all files, including the report (hint: under linux, `tar cvzf [name]_hwk[#].tar.gz [directory or files]`). **Please use the same question notation in your answer file**!

Not respecting these guidelines can result in a penalty of up to 10 points on your final score!

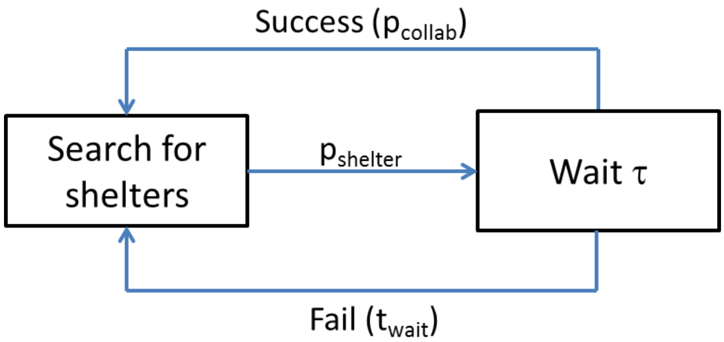
The number of points for each graded exercise is given between parentheses. **The combined total number of points for the graded exercises is 100.**

1.4 Before starting

To answer to the following questions, you will need to download **`hwk03.tar.gz`** from Moodle. Uncompress `hwk03.tar.gz`, containing the material needed to answer this homework. In addition, you will need to look at your solutions for the labs during this block.

1 Multi-Level Modeling of Robotic Swarms (30)

In this section we develop a mathematical framework using a simplification of the model of the collaboration experiment, similar to the model presented in [IJRR2004].



Here, a robot encounters a shelter with probability $p_{shelter}$ and thereby waits for maximally $\tau = t_{wait}$. With probability p_{collab} another robot enters the same shelter, and both robots revert to searching (the shelter is freed, and the success dance is omitted), yielding an average waiting time τ dependent on the probability of another robot coming to help.

Q_1(8): Have a look again at the Webots' collaboration model (*collaboration.wbt*) from Lab 7 (Section 4). Recall that in S17 you run the program for different waiting times and 4 robots. Now, run the program for different waiting times using only 3 robots.
How does the waiting time relate now to the collaboration rate? Why? Can you (approximately) guess a near-optimal waiting time using your microscopic model?

Q_2(5): How would the state machine change if three robots were needed for a successful collaboration? Assume now that the value of $p_{shelter}$ is the same regardless of the number of robots in a shelter (i.e., p_{collab} is no longer needed), and that each robot waits for a time of t_{wait}, resetting whenever a new robot enters the shelter. Draw the new state machine.

The rest of the questions in this section require three robots for a successful collaboration. Now, we assume that the PFSM from **Q_1** represents the whole team. Then we can think about the states as the number of robots in a certain state, i.e. $N_s(k)$, $N_{w1}(k)$ and $N_{w2}(k)$ as the number of robots searching, waiting alone, or waiting as a group of two, at time k respectively, with $N_0 = N_s(k) + N_{w1}(k) + N_{w2}(k)$ the total number of robots, and M_0 the total number of shelters.

Q_3(8): Derive: 1) the expressions $\Delta_{shelter1}(k)$ and $\Delta_{shelter2}(k)$ for the (average) number of robots that enter the state $N_{w1}(k)$ or $N_{w2}(k)$, respectively, and 2) an expression $\Delta_{collab}(k)$ for the (average) number of robots that *leave* a shelter after collaboration.

This allows us to formulate the following difference equation that describes the dynamics of one of the robot states:

$$N_s(k+1) = N_s(k) - \Delta_{shelter1}(k) - \Delta_{shelter2}(k)/2 + \Delta_{collab}(k) + \Delta_{shelter1}(k - t_{wait})\Gamma_1(k - t_{wait};k) + \Delta_{shelter2}(k - t_{wait})\Gamma_2(k - t_{wait};k) \quad (1)$$

Q$_4$(4): What do Γ_1 and Γ_2 represent? You can start your answer (one sentence for each) with: "Γ_x represents the fraction of robots that…"

Q$_5$(5): Our performance metric for the above system shall be the average collaboration rate

$$\overline{C}(k) = \frac{\sum_{j=0}^{k} C(j)}{k}$$

where $C(j)$ are the average number of collaborations at time j. How would you calculate $C(j)$ as a function of $p_{shelter}$, $N_s(k)$, $N_{w1}(k)$ and $N_{w2}(k)$?

2 *Distributed Task allocation* (37)

2.1 Heterogeneity in market-based systems

Consider a planetary exploration mission that needs to be accomplished by a team of robots. The robots need to either **take photos** of different sites or **extract samples** of soil. We want to distribute the robots such as to minimize the mission time.

Q$_6$ (6): Suppose that not all robots can perform all tasks. While some robots are capable of both taking pictures and extracting samples, others may only be able to perform one of the tasks. Will we have to change the global objective function from the one used in Lab 8 for the spill cleaning example, and if so, how? Will we have to change the local objective function, and if so, how?

Q$_7$ (4): Now, suppose that the robots are more arbitrarily specialized, i.e., each robot performs each task at a different speed. Consider that each robot R takes P_R seconds to take a photograph and S_R seconds to take a sample. Will we have to change the global objective function and, if so, how? Will we have to change the local objective function from the previous question and, if so, how?

Q$_8$ (4): Consider now that each robot has a limited capacity with which to store its samples and/or photographs. How does this affect the way that robots should bid?

2.2 Dealing with uncertainty

Let us continue with planetary exploration scenario and now consider various types of uncertainty. In all of these cases, assume that robots are pure specialists (i.e. they can either take photos or extract soil samples).

Q_9 (4): Suppose that we do not have full knowledge of the planet's surface, and that some areas may be covered by rock that the robots are unable to penetrate. Let there be a probability *p* that once a robot (of any type) arrives at a site, it is found to be unsuitable for sample extraction. In this case, the task should be ignored. How should robots bid on soil extraction tasks in this case?

Q_{10} (4): Building on the previous question, suppose that we would like to take a photograph of sites that we are unable to extract samples from. How should robots bid in this scenario?

2.3 Heterogeneity in threshold-based systems

I_{11} (15): Open the Webots world *i12.wbt*. This world is similar to the threshold-based example you saw in Lab 8, except that there are now two types of events which robots must handle. Currently, the robots only respond to the first event type. Modify the code so that the robots may adapt separately to both of the stimuli (i.e., *robots[r].stimulus[0]* and *robots[r].stimulus[1]*). Try changing the arrival rate of both event types using the constants at the top of *i12supervisor.c*. Under what conditions would such adaptation be useful? What goals might it help your robots accomplish? Implement an adaptation strategy which follows from your answer. You should submit *i12implement.c* along with this homework.

3 Collective Decisions (32)

The following questions refer to the collective decision scenario described in Lab 9. Assume global communication.

Q_{12} (5): Consider a system of 15 robots with a decision interval of one second. If you initialize each robot's state uniformly at random, what is the average time after which the system has reached a collective decision?

Q_{13} (3): Assume that right after initialization, 12 robots are doing right wall following and only three are doing left wall following. What is the probability that all robots will end up in a left wall following state?

In the collective decision algorithm used above, all robots converge towards doing the same task. Sometimes, one may wish to design a system that converges towards more complicated stable states. The following questions use local communication.

I_{14} (12): Open the Webots world *i15.wbt*. In this question, you have a system of ten robots. In *i15epuck.cc*, implement an algorithm with the following two absorbing states:

- 7 robots following the left wall, 3 robots following the right wall

- 7 robots following the right wall, 3 robots following the left wall

An absorbing state is one such that once the system reaches that state, it will no longer change. Your algorithm should make use of the fact that each robot is assigned a random unique ID in the range 1 – 200 (stored in the global variable *robot_id*). Other than this fact, the controller is identical to that seen in Lab 9. You should submit *i15epuck.cc* and a description of your approach.

Q_{15} (4): Assume we greatly increase the number of robots, but the number of IDs is still an order of magnitude greater than the number of robots. Does your above approach scale? Why or why not?

A scalable approach to the original problem might be to use a threshold-based task allocation algorithm. Assume that all robots occasionally broadcast the state which they are in (either "L" or "R") and that your robot therefore always knows n_L, the number of robots in state "L" and n_R the number of robots in state "R" within his communication range.

Q_{16} (5): What would be a sensible value for the stimulus s? What would be a sensible value for the threshold θ? Please indicate s and θ in terms of n_L and n_R.

Q_{17} (3): Does this approach meet all of the constraints proposed in **I_{15}**? Explain your answer.

3.1 References

[IJRR2004] **Modeling Swarm Robotic Systems: A Case Study in Collaborative Distributed Manipulation** / Martinoli, A.; Easton, K.; Agassounon, W. / *Int. Journal of Robotics Research*, Num. 4, Vol. 23 (2004), pages 415-436

To complete this homework, create an archive **[name]_hwk[#].zip** or **[name]_hwk[#].tar.gz**. This archive should contain a PDF report with the name **[name]_hwk[#].pdf** containing your answers and all the files that came with the original archive and are now modified. You should then submit this archive on Moodle.