*Alexandre SPRICH - Junyu LIN*

# TDDD07 Real Time Systems
# Lab 2 : Preparatory questions

1) **What is the role of the go-ahead message ?**

A go-ahead message ensures the continuous working of the robot : every second a go-ahead message will be received from the server, which means that the robot can keep working. But if a go-ahead message is not received, then the robot should stop until eventually a new go-ahead message is received.

2) **What are the different message types ?**

There are 4 types of messages that can be sent from the robot :
- **Victim reports** (when a victim is found)
- **Location messages** (which contain information about the position of the robot)
- **Pheromone maps** (which are shared among the robots to indicate which areas that have been searched)
- **Stream data** (which emulates live video transmissions)

3) **In which file(s) will solutions for lab 2 go ?**

Our solutions will be implemented in the files task_communicate.c , scheduler.c and task_mission.c.

4) **How can you make sure that the robot only sends messages within its allocated time slot ?**

Firstly we'll have to determine the priority of messages that the robot is going to send, put them in a queue then allocate the communication task in our schedule based on the given time slot.
Specifically, we start a timer at the beginning of the schedule, implement other tasks, wait until the given time slot comes and execute the communication task.
We can also know the maximum amount of data that can be sent in one time slot, by determining the bandwidth of the network. Thanks to this value, we'll be able to avoid overrun the time slot.

5) **What metric(s) do you plan to use in lab 2 ?**

We'll need the bandwidth of the WiFi network, the length of our messages, and the time of the time-slot allocated for the communication task.

6) **How do you plan to evaluate your solution ?**

## prioritizing and sending messages:

Firstly, we had to determine a priority order for the different message types :

1) Victim info AND robot position→ (critical) because victim info and robot position are important. Without the correct robot position, the robot won't receive go-ahead message
2) Pheromone maps→ (important)indicates which areas that have been searched, executed during the Navigate task
3) Stream data → (non critical) live video transmission

We built two doublelinked lists to store pheromone maps and stream data separately. After retrieving data from g_list_send, we used an if statement. If data type is critical info like victim tags or robot position, it will be packed and sent immediately. For pheromone maps and stream data, if the accumulative data sent for one time slot is below 2350 bytes, they can be packed, their size are checked again before they are sent. Pheromone maps has priority over stream data. Otherwise, the two doublelinked list is emptied. Finally, register "data" is freed.

The bandwidth of wifi network is 153600 bits per second (19200 bytes for 8 time slots), 2400 bytes per time slot. Leaving a margin of 100 bytes, we set the limit of sending data per time slot to 2300 bytes. We put a counter after each kind of data is sent as a UDP packet thus, we can print out the amount of these data transferred during one time slot.

## Checking go-ahead messages:

To measure how frequent the robot doesn't receive a go-ahead message, we add a timestamp as the schedule starts to execute and another timestamp when the robot doesn't receive a go-ahead message, by adding 1 every time a go-ahead message is missed, we can print out how many go-ahead messages are missing until the latest missing. We put the robot at several locations near those victim tags and started our measurement on checking go-ahead messages and how much data packages can be transferred within one time slot.