Birzeit University - Faculty of Engineering and Technology
Electrical & Computer Engineering Department - ENCS4330 - Midterm exam
Real-Time Applications & Embedded Systems - 2^{nd} semester - 2024/25 - duration: 120 minutes

Instructor: Dr. Hanna Bullata

Student Name/ID: ____

Drone clusters attack

We would like to simulate the behavior of drone clusters on missions to attack enemy targets. Some of the enemy targets are static (location-wise) and thus easier to attack and destroy while other targets are dynamic and thus harder to locate and destroy.

The following are ways to control clusters of drones while on missions:

- Each drone of a cluster can be controlled individually from ground stations and satellites. That is the hardest control way but you get maximum flexibility during missions and mission success rate is usually high.

- Controlling from ground stations and satellites only a subset of drones within a particular cluster (e.g. user-defined number of drones within the cluster). Usually the subset includes the master drone and few other drones. In that mode, the remaining drones of the cluster follow orders from these elected subset of drones. As an example, when that subset of drones receives an order to fly high, the remaining drones of the cluster follow the subset and all fly high. That control mode has medium success rate in fulfilling missions.

- Autonomous control mode. In that mode, the cluster is given the mission while on the ground and the cluster of drones is on its own to fulfill the mission. The success rate of that mode is mission-dependant. For example, if control is established with ground stations and satellites during the mission, the success rate is usually high.

Since the enemy might try to corrupt communication between drone clusters and ground stations or satellites, the mission plan must be uploaded to the cluster of drones just in case communication is interrupted during missions (see below for details). As such, we're mostly interested in the autonomous control mode.

The simulation can be described as follows:

- A cluster of drones is composed of a user-defined number of drones. Each cluster has a single master drone while the remaining ones are called slave drones.

  The master drone is responsible to make decisions (e.g. attack a specific target, cancel operation and go back to base, order slave drones to explode in a suicide fashion so as not to be captured by the enemy, change slave drones location within the cluster, etc). The master drone consults with the slave drones when making decisions though.

- While on a mission, a user-defined number of drones within a particular cluster has the complete plan of the whole mission. The master drone is always part of that subset of drones. If during a mission one of the subset drones (e.g. the master drone) is shot down or damaged for any reason, the subset of drones elect a new master drone as follows: Each will guess a random number in a user-defined range. The highest guess becomes the master drone. If 2 or more drones of the subset make the same highest guess, only those drones re-do a guess until only one drone becomes the master drone.

  Similarly, a drone not part of the subset must join the subset as a replacement for the lost master drone so all drones not part of the subset have to guess a random number in the same user-defined range. The highest guess joins the subset or the same iterative approach is used as previously described until a drone is elected to join the subset.

  When a drone joins the subset of drones, it is informed of the mission by one of the subset drones.

- During missions, and if no communication with ground stations or satellites, all drones of a particular cluster update their location according to data gathered from their own sensors. The drones make available their data (x, y, z, speed) to other cluster members.

The master drone needs to make sure all slave drones are within communication range (user-defined). Slave drones that are deviating need to be informed to adjust speed both ways by the master drone. If deviation persists, the master drone might instruct slave drones to abandon mission by returning to the base or exploding in mid-air (percentage is user-defined).

- If a slave drone loses communication with the master drone, it tries to re-establish communication by adjusting its position with the current predicted master position. If that doesn't resolve the issue, a slave drone tries to communicate with any drone within the cluster. If that succeeds, the intermediate drone informs the master drone about the location of the slave drone that lost communication so that the master drone decides on what to do (e.g. slow down the whole cluster). If that fails, then the slave drone can turn back to the base or explode in mid-air. The probability of doing either or is user-defined.

- If the cluster of drones is targeting a static target, then the master decides if the target location is reached or not yet. However, if the target is dynamic and has no specific location, the master drone informs all other drones to search for that target. If the dynamic target has been identified by one drone of the cluster, the remaining drones are requested to confirm that fact. If more than a user-defined number of drones confirm locating the target, the master drone might order to launch an attack on that target. Otherwise, no attack will take place.

- When drones are in attack mode for static targets, the drones of the cluster follow the master drone in terms of altitude and speed. For maximum destruction effect, reaching the target has to happen simultaneously.

  While attacking dynamic targets, the drone cluster follows the drone that has maximum identification of the target. As such, and while in attack mode, the drones adjust their altitude and speed dynamically to follow the drone that has maximum identification of the target.

- The simulation ends if any of the following is true:
  - The number of master drones that have been destroyed during the mission exceeds a user-defined threshold.
  - The number of drones that went back to the base exceeds a user-defined threshold.
  - A cluster of drones exploded in mid-air in a suicide fashion.

What you should do

- We would like to implement the above problem on Linux machines using a multi-processing approach.

- Using a tabular format, describe in human language how you would implement the above described problem. Name the file `drone_cluster_description.txt`. Discuss the IPC techniques you would employ for the simulation.

  The number of lines in the description file must be reasonable (50 lines max).

  If you use chatgpt tool or other LLM tools to generate the description, you'll get a zero! **(10 points)**

- In order to avoid hard-coding values in your programs, create the file `config.txt` that contains all the values that should be user-defined. **(5 points)**

- Using C-language, write the code for the drone <u>only</u> in a file called `drone.c`. **(15 points)**

When the exam time is over, send the files:

- `drone_cluster_description.txt`,
- `arguments.txt`,
- `drone.c`

as a reply to my ritaj memo entitled **encs4330 midterm exam - May 31, 2025**.