# Task Sheet 1 - Scalability and Synchronization

**Objectives:**

- Get acquainted with scaling of response times
- Investigate a method to synchronize an asynchronous swarm
- Experience influence of swarm connectivity/swarm density

**Deliverables:**

- Please form **groups of max. two**.
- **For each submission include:** source code, plots and a readme of how to compile & start, and a characteristic diagram for each subtask where applicable
- **allowed tools:** C/C++, python, Java, matlab (or ask me before)
- Please zip your submission in a single file named:
  "collRob_sheet1_YOURLASTNAME1_YOURLASTNAME2.zip"
- Solutions will be presented in class in person. Both group members must take part in the presentation.

# 1 Scaling of a Data Center

We create a simple model of a data center. The following is a simple example from queuing theory. We assume that new data for the data center come in with a constant rate of $\alpha$ and are added to a waiting list. We also assume that the process is without memory, that is, incoming data are statistically independent from each other. Therefore, we model the incoming data as a Poisson process. The probability that $i$ data arrive within a given time interval $\Delta t$ is given by

$$P(X = i) = \frac{e^{-\lambda}\lambda^i}{i!} \quad , \tag{1}$$

for $\lambda = \alpha\Delta t$. In the following we simply set $\Delta t = 1$.

A. Plot $P(X = i)$ for a reasonable interval of $X$ and $\alpha \in \{0.01, 0.1, 0.5, 1\}$.

B. Implement a program that samples numbers of incoming data from $P(X = i)$.

C. Implement a model that iterates over these two phases: calculate and administrate new incoming data, then operate on the current data (one at a time). Generate a sample of 2000 time steps for $\alpha = 0.1$ and a processing duration of 4 steps per data. What is the average length of the waiting list?

D. Change your program such that you can average the waiting time list length over many samples (independent runs of your model over 2000 time steps each). Determine the average list length for rates $\alpha \in [0.005, 0.25]$ in steps of 0.005 based on 200 samples and plot it.

E. Do the same for a processing duration of only 2 steps per data with rates $\alpha \in [0.005, 0.5]$ in steps of 0.005. Compare the two plots.

## 2 Synchronization of a Swarm

Figure 1: Synchronized flash of fireflies.

Swarm systems are asynchronous systems. There is no central clock that could be accessed by everyone. If a swarm needs to act in synchrony it has to explicitly synchronize first. An example of a biological system that shows synchronization is a population of fireflies. "Though most species of firefly are not generally known to synchronize in groups, there are some (for example, *Pteroptyx cribellata*, *Luciola pupilla*, and *Pteroptyx malaccae*) that have been observed to do so in certain settings."

In the following we create a simple model of such a firefly population. The population is scattered randomly over a square of 1 by 1 (uniform distribution). We assume that fireflies are stationary and can only perceive local neighbors in their vicinity. We say two fireflies are within the vicinity of each other if the distance between them is smaller than $r$. Hence, there is a virtual disc centered at each firefly's position and every other firefly sitting on that disc is a neighbor. The fireflies flash in cycles. We define the cycle length by $L = 50$ time steps. We initialize the fireflies to uniformly randomly distributed clock cycles (i.e., not sycnhronized). The firefly flashes for $L/2$ steps followed by $L/2$ steps of not flashing. This holds except for those cases when the firefly tries to correct its cycle to synchronize. In the time step after it has started to flash it checks its neighbors and tests whether the majority of them is actually already flashing. If so, the firefly corrects its clock by adding 1, that is, it is decreasing the current flashing cycle from $L/2$ to $L/2 - 1$ steps and will consequently flash 1 step earlier next time.

A. Implement the model for swarm size $N = 250$ and cycle length $L = 50$. Calculate the average number of neighbors per firefly for vicinity distances $r \in \{0.05, 0.1, 0.5, 1.4\}$. Plot the number of currently flashing fireflies over time for vicinity distances $r \in \{0.05, 0.1, 0.5, 1.4\}$ for 5000 time steps each. When plotting the number of currently flashing flies, make sure you plot the full interval of $[0, 150]$ for the vertical axis.

B. Extend your model to determine the minimum and maximum number of concurrently flashing fireflies during the very last cycle (last $L = 50$ time steps starting from $t = 5000$). By subtracting the minimum from the maximum you get double of the amplitude of the flash cycle. Average the measured amplitudes over 50 samples each (50 independent simulation runs with 5000 time steps each) and plot them over vicinities $r \in [0.025, 1.4]$ in steps of 0.025. What seems a good choice for the vicinity and the swarm density?

# References


A. Merkle, Daniel, Martin Middendorf, and Alexander Scheidler. "Swarm controlled emergence-designing an anti-clustering ant system." 2007 IEEE Swarm Intelligence Symposium. IEEE, 2007.

B. Scheidler, Alexander, Daniel Merkle, and Martin Middendorf. "Swarm controlled emergence for ant clustering." International Journal of Intelligent Computing and Cybernetics (2013).