Bio-Inspired Artificial Intelligence - EMATM0029 Exercises - Bio-mimetic Robot Control

The sections ***Digging deeper** are optional. These are for the ones who want to learn more about the topic.

Central Pattern Generation

TED talk

Watch Auke Ijspeert's recent TED talk on his work with CPGs. [link]

Matlab

Download the Matlab code from Blackboard to simulate coupled CPGs for a quadruped robot. Have a look at the code `cpg.m` it implements 4 coupled Hopf oscillators.

1. Start the simulation with `simulatecpg.m` a couple of times to see how the system copes with random starting points.
2. Change the range of possible initial starting points to values between 0 and 10 by changing the code to `y0 = 10*rand(1,8);` Try also negative values. Why does the system behave like this?
3. Change the connectivity matrix `K` in `cpg.m`. It defines how the CPGs influences each other. Use only values 0,1 or -1. How does the output change?
4. Change the elements of the connectivity matrix `K` to any values of your choice. How does this change the output?

*Digging deeper

- If you want to see more applications of CPGs have a look at the BioRobotics webpage [link].
- Adapt the Matlab code such, that at a certain point in time (e.g. $t = 10$ sec) one CPG switches off (i.e. $x = 0$ and $y = 0$). How does the system react to that?
- Use the same setup, now with the switched off CPG recovering (switch on again, e.g. at $t = 20$). Is the whole system able to recover?

Subsumption Architecture

Imagine you're designing the controller of the Roomba Vacuum cleaner (designed in part by Rodney Brooks - the creator of the subsumption architecture). The robot has two wheels, a bumper at the front, and a wall sensor on the right of the robot.

Draw the subsumption controller that you believe would give you the best area coverage, given a square room with a sofa, a table, and 4 chairs.

How would you change your subsumption controller to work on the Dyson 360 eye, which is equipped with 360 vision?

What are some of the pros and cons of using a subsumption architecture, compared to less reactive/more plan-based controllers?