**Acquired Intelligence & Adaptive Behaviour Simulated Robots**
**Lab Session 6**

**Goals**:
1. To extend your investigation of a simulated low-inertial wheeled robot by implementing a single-sensor photo-taxis (light-following) algorithm.
2. To optimises phototaxis using random search, a hillclimber or a GA.

**Background**:
In the previous lab session a simple low-inertia wheeled robot was introduced, to investigate how four of Braitenberg's hypothetical 'Vehicles' might be implemented in simulation. In this session you will be challenged to modify the wheeled robot to perform photo-taxis (light-following) using only one sensor and to report sensible statistics on its performance that could, for example, be used as the fitness function in a genetic algorithm.

In a second task, using either random search, a hillclimber or a GA (from previous labs) we will attempt to optimise efficient phototaxis. To do this you will need to build a fitness function and a mutation operator.

**Tasks:**
**A.** Implement a controller that enables photo-taxis with only one sensor.

For the first task you will need the files init_sim.m and run_sim.m.
1. To do this you must first lesion (cut) your agent such that one of its sensors doesn't work. You should then think about how you might go about setting up the neural controller to solve the task. Note that we only need the agent to get reliably closer to the light source over time - there is no need for explicitly directed behaviour (although this is possible).
Tip: Write down your ideas - you may find that explaining ideas in words is much easier than implementing them in the simulation!

2. If you think you have a decent solution, you should now be able to prove it. Can you devise a way of quantifying the performance of your agent at the photo-taxis task? Run the simulation multiple times (with different starting conditions) to get some idea of how the the agent performs in a range of environments. Compare these statistics between two or three alternative implementations.

**B.** Optimise phototaxis behaviour.

For this task you will need your hillclimber or GA code from previous labs and the simple_agent.m provided on study direct.

1.Open simple_agent.m and examine how it works. It takes a genotype which specifies the weights and biases of an agent as a genotype, i.e., [w_ll w_lr w_rl w_rr bl br]. It runs an agent for T time steps and returns the trajectory of that agent. Examine the behaviour of some randomly constructed agents.

3. Construct a mutation operator. For a real valued genotype you should typically add a random value to each gene which is about 1% of that genes range, e.g. here for weight we add randn*2/100 and for biases randn*1/100 to each gene. Remember you must disallow weights or biases that go out of range. Do agents separated by a single mutation act similarly? What do you think the fitness landscape is like?

3. Construct a fitness function. Using the statistics you developed in task **A** construct a fitness function that takes the trajectory returned by simple_agent.m and computes a single value which is proportional to the performance of your agent. Remember to confirm you fitness function visually.

4. Now start agents at position [10,10] with a random bearing. Using your fitness function and mutation operator use either random search, a hillclimber or your GA code from last week to find the optimal weights and biases for phototaxis from these start conditions. The best agents will go directly to the light and stay in its vicinity thereafter.