2019

Simulated Robots

AIAB REPORT

168908

Contents

Simulated Robots 2

- Abstract 2
- 1 Introduction 2
  - Braitenburg Robots 3
  - Phototaxis 3
- 2 Method 4
  - 2.1 Optimising Robots 5
- 3 Results 6
  - 3.1 Aggressor: Contra-lateral, positive connections 6
  - 3.2 Coward: Ipsilateral, positive connections 7
  - 3.3 Lover: Ipsilateral, negative connections 8
  - 3.4 Explorer: Contra-lateral, negative connections 9
  - 3.4 Two-wheeled, single sensor positive connection robot 9
  - 3.5 Microbial Genetic Algorithm generated robots two sensors 10
  - 3.6 Microbial Genetic Algorithm generated robots one sensor 12
  - 3.7 Effect of the different starting locations 14
- 4 Discussion 16
- 5 References 17

Simulated Robots

Keyword: Braitenberg, embodiment, phototaxis, ipsilateral, contralateral, excitatory, inhibitory

Abstract

Intelligence was thought to be a very complex behaviour and believed to be the result of a vast assembly of nerve cells and their associated molecules (Crick 1994). In this study, I will be building simulated robots to see how the role of body and environment affects the behaviour of the individual. For example, an ant can be viewed as a behaving system where the ant itself is quite simple, but the complex behaviours exhibited is largely a reflection of the environment (Simon 1969). To investigate the complex behaviours exhibited from simple organisms, I will be using the Braitenberg Vehicles to help investigate and support my studies. Also using the help of genetic algorithms to find the optimal solutions to a given task.

Hypothesis: Complex behaviours can be explained by simple reactions to the environment. And the parameters found by the genetic algorithm should go to the light faster than that found by myself. Furthermore, I expect two sensors robot to be able to find the light faster than the one sensor robot. If there is more than one light, I expect the robot to converge to the light closer to itself.

1 Introduction

Often, we associate complex behaviours with the very sophisticated biological mechanism behind it. It is very logical for us to assume that when animals display organisation and almost impossible tasks unless for an intelligent organism. Through our understandings in biology, we think there are very complex brain activities made by neurons and synapses are the cause for complex behaviours. Simon (1969), looked into animals and their behaviours in the book "The sciences of the artificial". He decided to use Braitenburg robots to study how seemingly complex behaviours can be caused by very basic internal structures known as emergence (Brooks 1995), the robots are very simple and only features wheels and light sensor.

With the rise of computational power, it is possible to implement a simulated environment in which Braitenburg robot could be tested, it is better to do this study in the simulated world because all the variables are controllable, and the environment will not include noises which could be present in the real world. I will make several Braitenburg robot that displays some behaviours that are often seen in nature, such as aggressive behaviour, loving behaviour (attraction), coward (running away) and explorer (visiting many areas).

Simulations are good for doing this study because simulations run on the computer offers faster speed when trying to obtain results and it is easily able to be repeated many more times than it is practical with a real robot. In the virtual environment, it is very good for implementing new features because the virtual space is unconstrained so in my study the environment could be changed. Alternatively, I could make the environment very simple and free from external factors like the real world.
In the previous study, I have designed and built genetic algorithms which solves an optimisation problem. The genetic algorithm used is a biologically inspired algorithm that gets ideas from

microbial reproduction and uses evolution to find better genotypes. In this study on simulated robots, I will implement the genetic algorithm to find the best genotype for the robot to reach the light. The ability of the robot to evolve and use sexual reproduction to improve its performance is mentioned by Eiben(2013). Please see the attached code to see the algorithm used for part of this study.

I found many interesting behaviours by implementing the various Braitenberg vehicles in the simulated environment, such as how the connections between sensors and motors affects the behaviour of the vehicle completely and how small changes to the values can show very different behaviours. Also discovering that using genetic algorithm to find the optimal parameters for the simulated robot is much better than my hand designed parameters.

Braitenburg Robots

As shown in figure 1, the Braitenberg vehicle would have wheels and light sensors. The models shown in figure 1 have two wheels and two light sensors each, they are different by the connections between the wheel and the lights. In this study, I will be using such diagrams to show the type of structure of each type of vehicle.

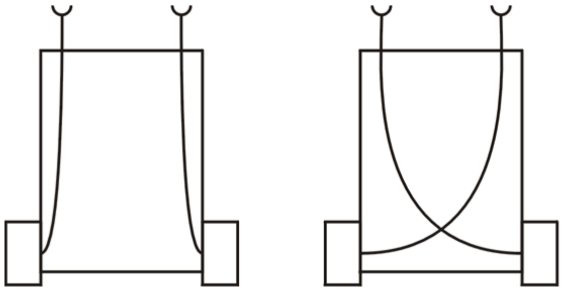

Figure 1- illustration of Braitenberg vehicles, left vehicle is Ipsilateral and the right vehicle is Contra-lateral

Phototaxis

Phototaxis is known as a locomotory movement which is the movement of the entire organism due to the presence of light, this phenomenon is seen with many animals in the natural world. Cockroaches run to the place with lower light intensity, this almost immediate response when light is turned on is a sub-conscious action as cockroaches don't seem to look around and decide, it is a rather immediate action towards the direction of lower light intensity. This is a negative phototactic response shown by the cockroach and this can be explained by biology that it is safer for cockroaches to stay hidden in the darkness and may have been a reflex reaction developed from evolution.

In contrast, the positive phototactic animals like the moths, these insects are always going towards the source of light. This is advantageous for phototrophic organisms as they can position themselves to efficiently go towards the source of the light. Moths are, therefore, a positive phototactic animal in that its movement is in the direction of increasing light intensity.

In my simulated Braitenburg robots, the phototaxis is represented by having the light sensor/sensors changing the speed of the individual wheels. This would allow phototaxis to be displayed when a source of light is given.

2 Method

The implementation of these Braitenburg vehicles is done in MATLAB and the basic Braitenburg robot (vehicle) code is in the file simple_agent.m. This code takes in the time the vehicle will run for in ‘T’, and the position of the start for the vehicle given by ‘pos’, a bearing in which the vehicle will face towards in ‘bearing’, and a genotype of values each representing the weights of the sensors to the wheels .

The fitness of this vehicle will be the final distance away from the light source, it will measure how close the vehicle gets to the light source and it is used to compare the performance values when changing the genotype.

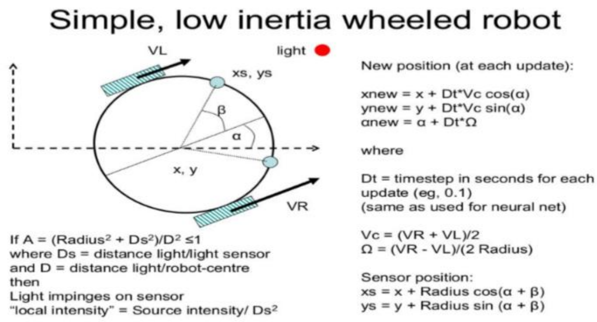

Figure 2 shows the simulated robot's mathematical details, explains how simple_agent runs

In figure 2, simple, low inertia wheeled robot used in this study is described as follows. The control architectures of these robots are considered ‘representation free’ where the motors are connected to the sensors via direct innervation, which can be excitatory (positive) or inhibitory (negative) and could have a bias (the last two values in the genotype). The wheels and sensors could be ipsilateral (same side sensor connected to same side wheel) or contra-lateral (the left sensor connected to the right wheel and right sensor connected to the left wheel).

I will use the simple robot in figure 2 to investigate which simple mechanism allows complex behaviours such as aggressiveness, cowardness, lovingness and exploration to occur. Then explore if the robot could do similar complex behaviours with only one sensor. To improve the result of the study, I will use a genetic algorithm to try to find the most optimal values for the robot to display phototaxis behaviours and testing the effect of distance that will have on various vehicles and additionally, how the bearing could affect the vehicles’ performances.

Furthermore, I will experiment with a single sensor robot with both one wheel, shown in figure 3 and two wheels and see their performances and behaviours in the environment. In addition, the connectivity will be tested to see the effect of a positive and negative connection between sensor and motor.

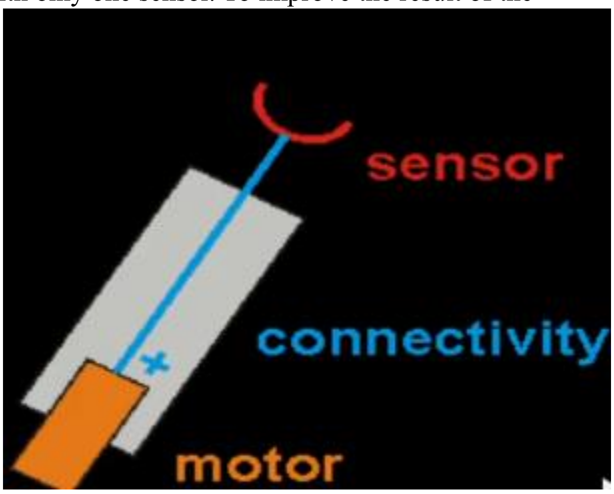

Figure 3 Single wheeled, single sensor positive connection robot

2.1 Optimising Robots with microbial genetic algorithm

I will use the full microbial genetic algorithm (Harvey 2001), this algorithm attempts to generate multiple genotypes, in this case, the different variables for the simulated robots to act upon. The various genotypes will be run and a resulting fitness (how close they got to the light) will be returned, this algorithm then picks out two random individual genotypes and compares their fitness for winner and loser, the loser gets a certain probability of values (genes) from the winner and is mutated and placed back in the genotype pool for the algorithm to repeat. The full microbial genetic algorithm attempts to find the most optimal values for the robot to get the highest fitness value, so this is better than finding it manually. The values for the genes are generated between -4 and 4, if the gene is mutated as the loser then a random mutation will be added to the chosen gene at around 1% of the original value.

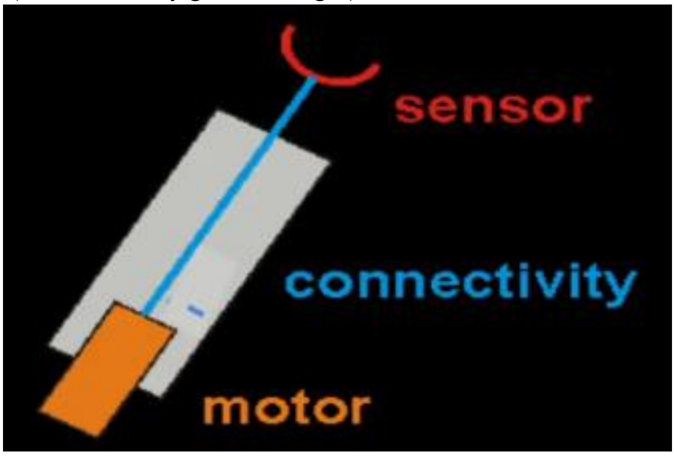

Figure 4 Single wheeled, single sensor negative connection robot

If the value exceeded the range then it will be set to either the maximum value or the minimum value depending on which is closest.

The Pseudo code for Full Microbial Genetic Algorithm

1. *Initialise random population P*
2. *Associate each individual with a position p*
3. *Pick one individual at random to make it p1*
4. *Pick a second individual G2 in the local neighbourhood of the first according k*
5. *Compare p1 and p2 finding a winner (W) and loser (L)*
6. *Copy each gene of the winner W to the L with crossover (PCrossover)*
7. *Add a mutation according to probability (PMutate) to the L*
8. *Until success or give up, go to 3*

For the optimising algorithm to function, there must be a way to measure its fitness value. The method I used in my study is to use the negative sum of the distance of the agent over time to quantify a number for the algorithm to improve on. The formula in figure 5 is used to work out the distance between the agent and the light with Pythagoras and to make it so that the genetic algorithm could attempt to optimise the parameters, the negative sum is used because unless the fitness is of a negative value it is not possible to try to get the distance smaller. As smaller distance is better fitness for this study.

$$-\sum_{t=0}^{t=T} \sqrt{(x_t^2 + y_t^2)}$$

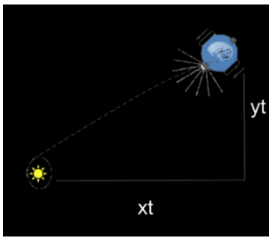

Figure 5 negative sum of the distance of the agent to light over time, it is a measure for fitness and it is negative so that the genetic algorithm made in the previous study could be implemented to optimise.

The actual code in MATLAB is attached to this study for reference.

3 Results

3.1 Aggressor: Contra-lateral, positive connections

The aggressor is a contra-lateral, positive connection. This allows the agent to start off away from the light at (1, 1) and at 90 degrees bearing. To make it **contra-lateral** the 2nd and 3rd number in the genotype is given the positive weights because those two values represent the left sensor to the right motor and right sensor to the left motor. The result is shown in figure 6 where the robot (blue) heads towards the light (yellow). This behaviour works by the left sensor detecting more light than the

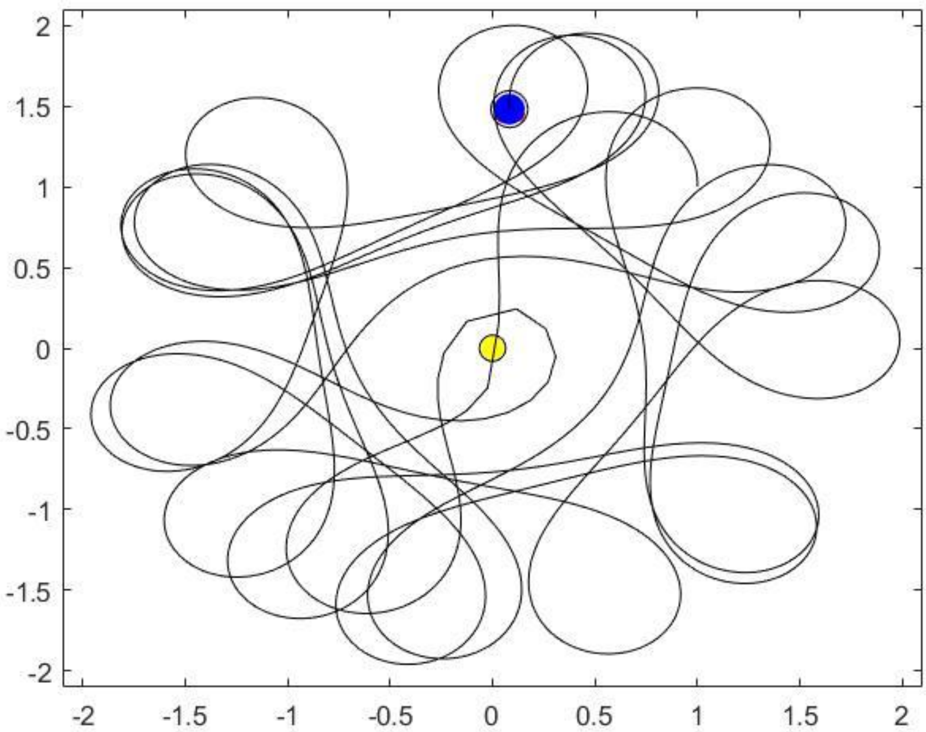

Figure 6 Aggressor: Contra-lateral, positive connections. simple_agent(100,[1; 1],90,[0 0.9 1 0 0 0.2],true);

Robot in blue is going towards the light and after going past the light, it is able to change direction to return to the direction of light. This is the behaviour known as aggressor mentioned by Braitenberg.

right sensor so the corresponding right wheel spins faster causing the robot to head to the left towards the light. (note: only describing this particular robot in figure 6) I discovered that there must be a slight difference between the weight values to allow the robot to go back towards the light once it has overshot in a direction, otherwise once heading away from the light, the weights need to be slightly different for the robot to turn around. The bias on the right wheel is to allow the robot to get closer to the light because the bias of 0.1 was displaying similar behaviours but did not appear to get as close to the light as when the bias on the right wheel is set to 0.2.

3.2 Coward: Ipsilateral, positive connections

The coward is the behaviour of going away from the light, this can be demonstrated with positive weight values on the 1st and 4th genotype values because they represent left sensor left motor and

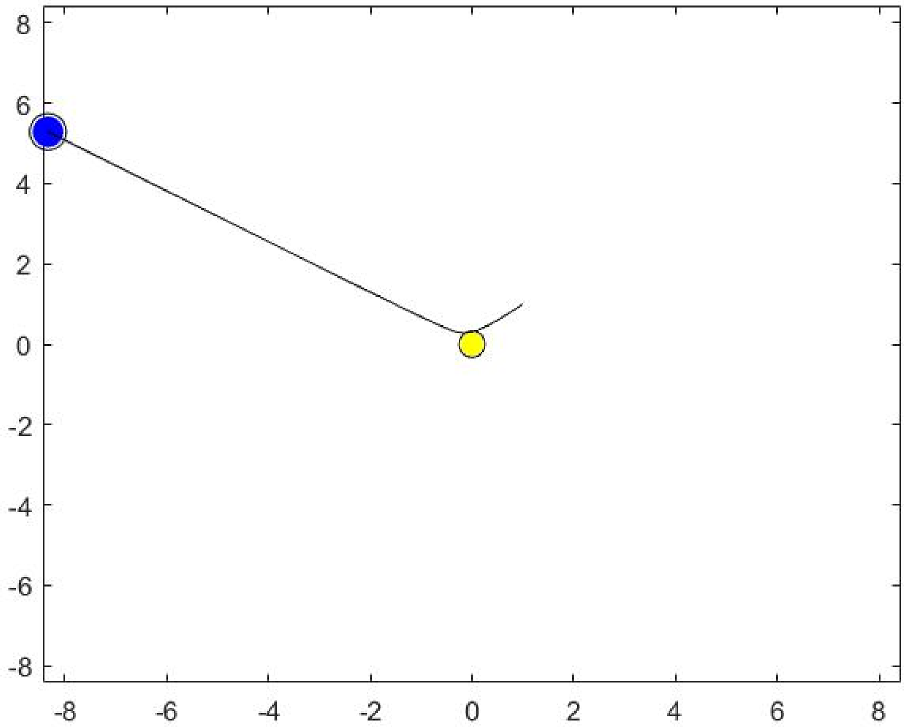

Figure 7 Coward: Ipsilateral, positive connections, simple_agent(50,[1; 1],220,[1 0 0 1 0 0],true);

It can be seen that the robot was facing the light at the start but as it got closer to the light, it went to the direction away from the light and the speed of the robot can be seen increasing after it faced away from the direction of the light. This is the behaviour of a coward described by Braitenberg.

right sensor right motor. The result is shown in figure 7. This shows the robot going further away from the light rather than going towards it. This behaviour can be explained by the stronger light sensor values on the right causes the robot to turn left because the right motor is going faster hence the robot will behave like a coward. This simple set up between the sensor and motor of the same side (ipsilateral) can show the behaviour of a coward, it matches with the behaviours with animals of the real world such as cockroaches.

3.3 Lover: Ipsilateral, negative connections

The lover is predicted to go towards the light and stop, this is because it features ipsilateral connected

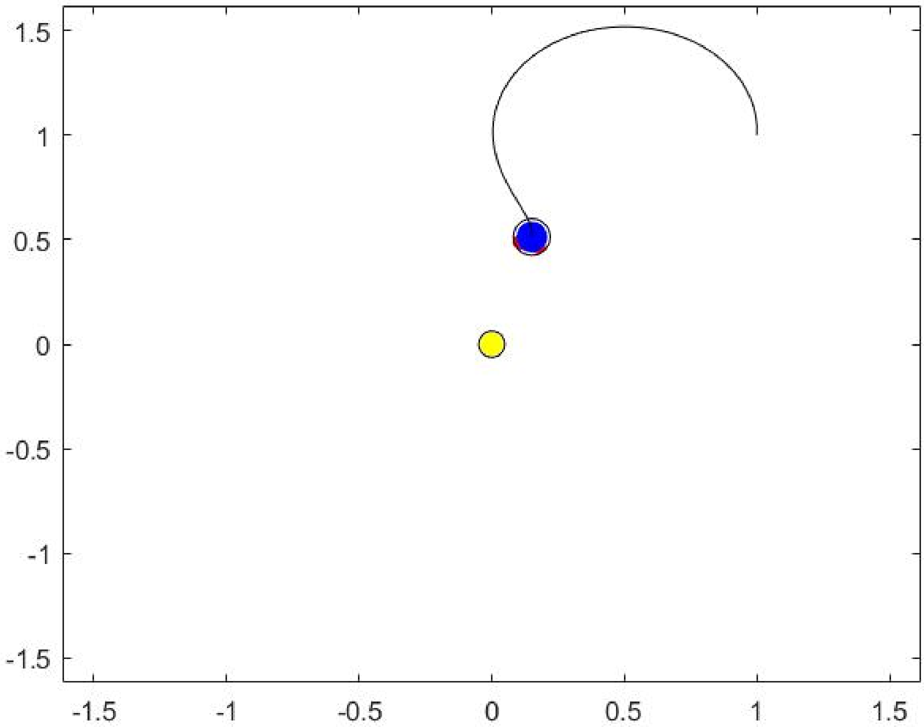

Figure 8 Lover: Ipsilateral, negative connections, simple_agent(100,[1; 1],90,[-0.5 0 0 -0.6 1 1.2],true);

The robot can be seen initially heading up and it turns to the direction of the light and once it has readjusted itself at the direction of the light, it slows down. I used T = 100 and it can be seen that it stops moving once it got close to the light. This is the behaviour of a lover described by Braitenberg.

sensors and motors in that they must go towards the direction of stronger light and connection being inhibitory, the motors have a bias to keep them on but will stop due to the negative connections. The 1st and 4th genotype values mean it is ipsilateral and negative values of -0.5 and -0.6 respectively, and the biases of the motors as positive values of 1 and 1.2 respectively. The negative weights on the sensor to motor weights are to allow the robot to stop getting closer to the light and the positive biases are to allow the robot to go towards the light when the light intensity values are not so high and the slight difference between the values is to allow the robot to navigate to the direction of light. When the sensors detect more light, the resulting values from the weight of sensor to the motor will be greater negative value than the positive biases, resulting in the vehicle’s motor values being below positive value so the vehicle will stop hence showing the lover behaviour.

3.4 Explorer: Contra-lateral, negative connections

The exploratory robot features contra-lateral negative connections with positive biases. The robot will slow down near the light as the sensors detect more light it will slow down because the biases are having fewer effects on the motors. And the robot will turn to head off in another direction because once the sensors are facing away, the motors will have more speed due to the biases.
Time was set to 50 to see the robot display its exploratory behaviours and the 2nd and 3rd genotype values are set as -1s to represent the negative connections and the last two genotype types set to positive 1 as the biases.

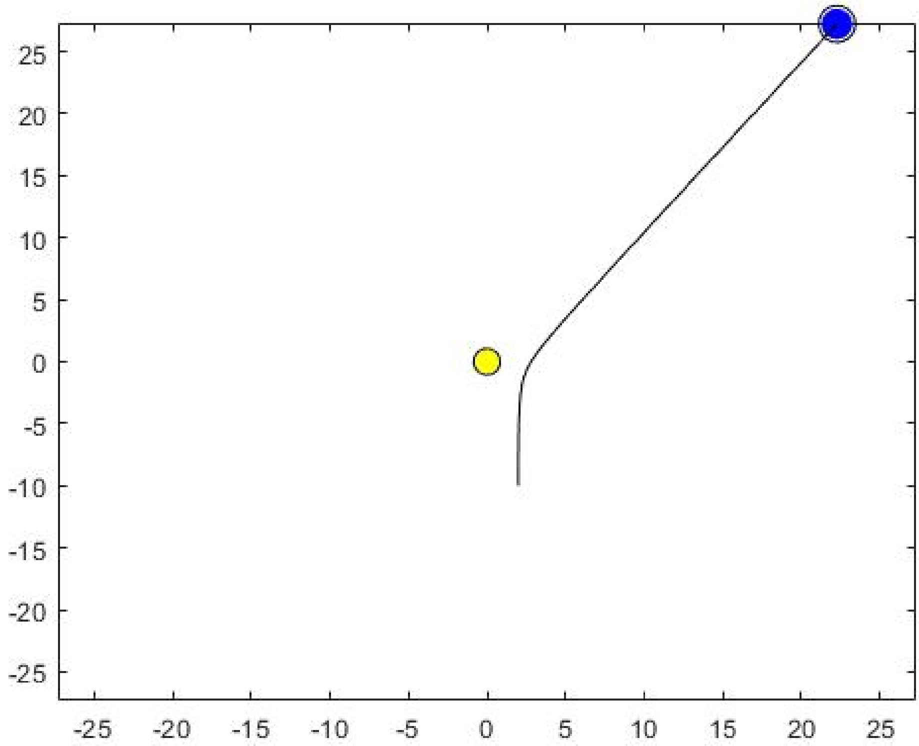

Figure 9 Explorer: Contra-lateral, negative connections, simple_agent(50,[2; -10],90,[0 -1 -1 0 1 1],true);

It is clearly visible that the robot got closer to the light and slowed down before heading away from the light. This demonstrates the exploratory behaviours of the robot.

3.4 Two-wheeled, single sensor positive connection robot

For the simulated robot to have only one sensor, it is possible to not put any weight values on the chosen sensor. In this case, I chose to disable the left sensor, and this is done by setting the w_ll and w_lr as 0 so when the value gained from the sensor on the left is multiplied by the relative light values. The resulting motor speed is not affected by the left sensor's value because when weights are set as 0 then the product must be 0 too. The 1st and 2nd genotype values are set to zero and 3rd and 4th are set as 1 and 1.2 respectively, this two non-zero values will make the robot react to the values obtained by the right sensor. Adding bias to the right motor will make the robot go in circles and the idea is that when the robot is facing the light, the increased light intensity will make the wheels spin faster towards the light hence achieving the goal of going closer to the light.

As shown in figure 10, the robot managed to get closer to the light and passing the light, despite after going past the light, it is evident to see the robot continuing to circle the light.

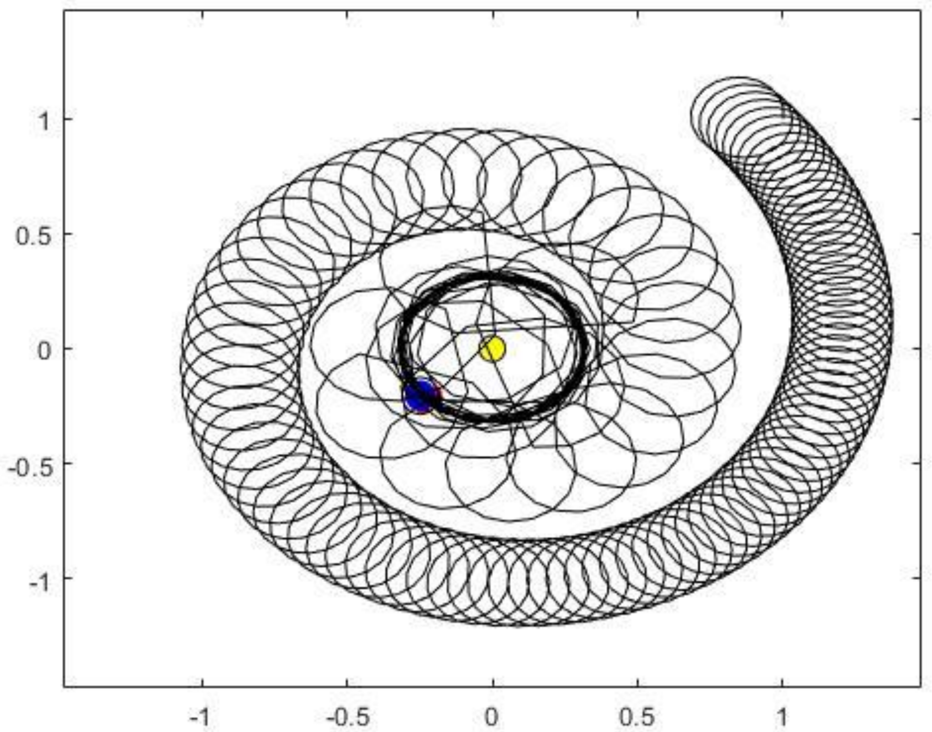

Figure 10 Single sensor two wheeled robot with a bias simple_agent(1000,[1; 1],90,[0 0 1 1.2 0 0.5],true);

the single sensor robot manages to get closer to the light because as it spirals in circle, when it is facing the light, both wheels will spin faster so for every spiral, it edges closer to light

3.5 Microbial Genetic Algorithm generated robots two sensors

Using the full microbial genetic algorithm mentioned in the method section, I found some surprising results and the best robot agent is shown in figure 11. It can be seen that the robot starts off in the position (1, 1) and starts to spiral towards the light and the spiral can be seen to change shape as it approaches nearer to the light eventually performing some seemly odd moves around the light at close proximity. Upon closer inspection, the robot seems to reach the light source and overshoots in the direction before quickly going back towards the direction of the light. The scale value suggests that the robot is extremely close to the light and stays close which is as close as 0.1 units away.

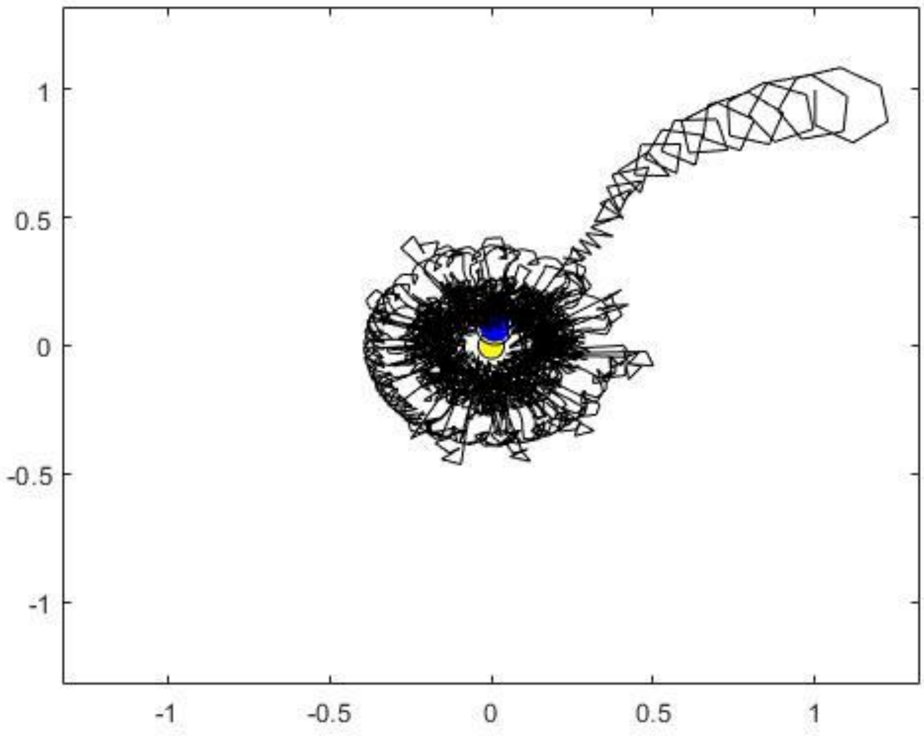

Figure 11 Simulated Robot from Microbial Genetic Algorithm, demonstrates how the microbial genetic algorithm manages to find a good solution by randomly comparing the set of random genotypes it produced in the population and through mutations, the best solution is displayed.

In figure 12, the initial parameters chosen randomly by the microbial genetic algorithm starts off having a low fitness level and it is around 80th iteration that the genotype (parameters) managed to find a better solution to the problem and it continues to find better solutions to the robot. This demonstrates the power of utilising computational power to find a better solution for the robot. Perhaps with more generations, the even better result will be found, and it is evident that the solutions by the robots are drastically different from the manually designed parameters.

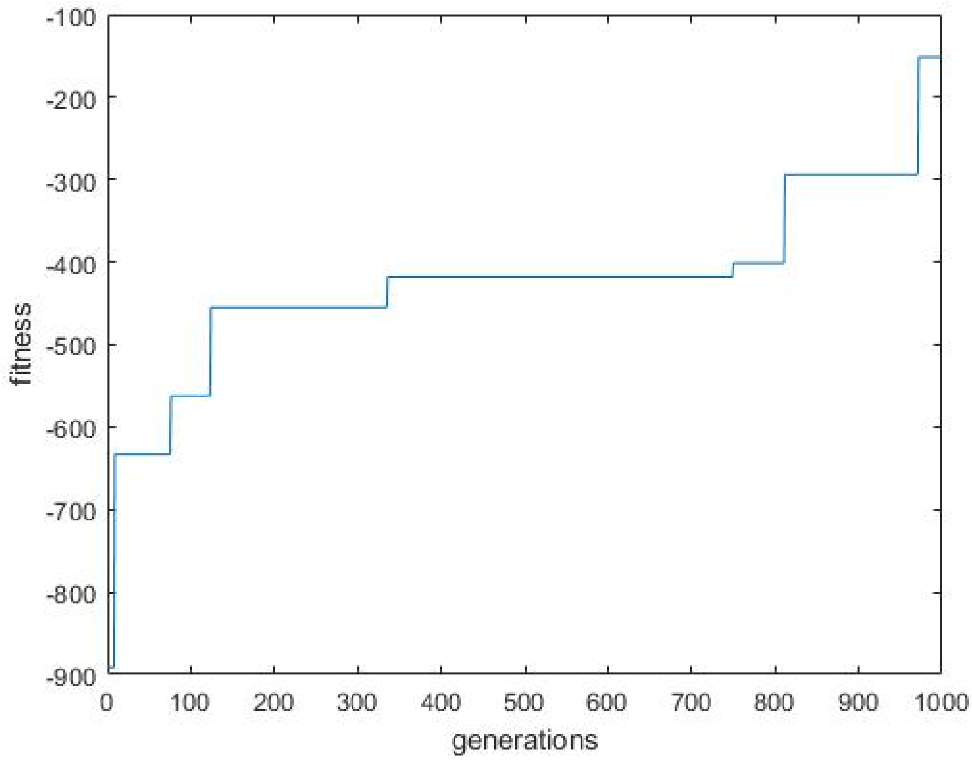

Figure 12 fitness against 1000 generations graph for microbial GA robots, it can be seen that the fitness growing over the generations and eventually getting very close to the light (closer to 0 in fitness is closer to the light)

3.6 Microbial Genetic Algorithm generated robots one sensor

For the genetic algorithm to be able to work out the best parameters for a single sensor robot. I changed the code slightly so that the genotypes are reset to zero for the first and second genotype as they are the values for which the left sensor is responsible for. Hence, rendering the left sensor inoperative and thus making it a single sensor robot. The best solution is

0 0 -0.4632 0.7054 -2.2668 -2.5517

the algorithm did put negative biases of varying values and the weights on the wheels to be both positive and negative.

In figure 13, it can be seen that the robot initially spirals and as it gets closer, the spiral size decreases until it reaches very close proximity to the light source and eventually reaching extremely close to the light.

Figure 14 shows the ability of the algorithm to find better solutions and eventually reaching a really good fitness value of close to 0. This demonstrates that a single sensor robot actually performed better than with two sensors.

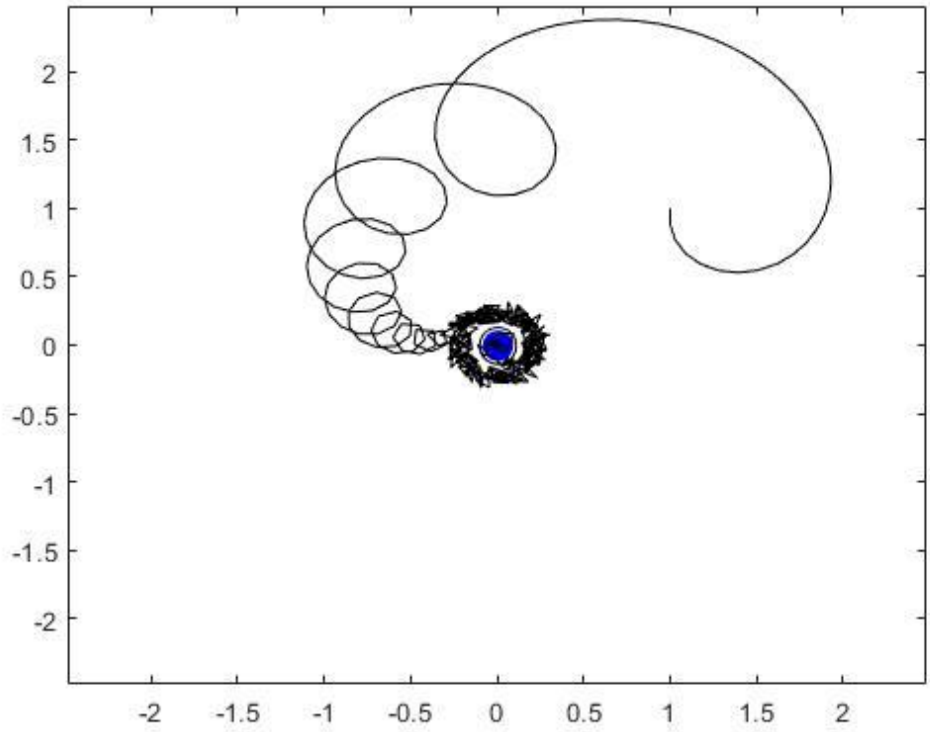

Figure 13 single sensor robot from Microbial Genetic Algorithm, it shows the single sensor robot generated from the genetic algorithm is very capable in reaching the light using the familiar spiral approach.

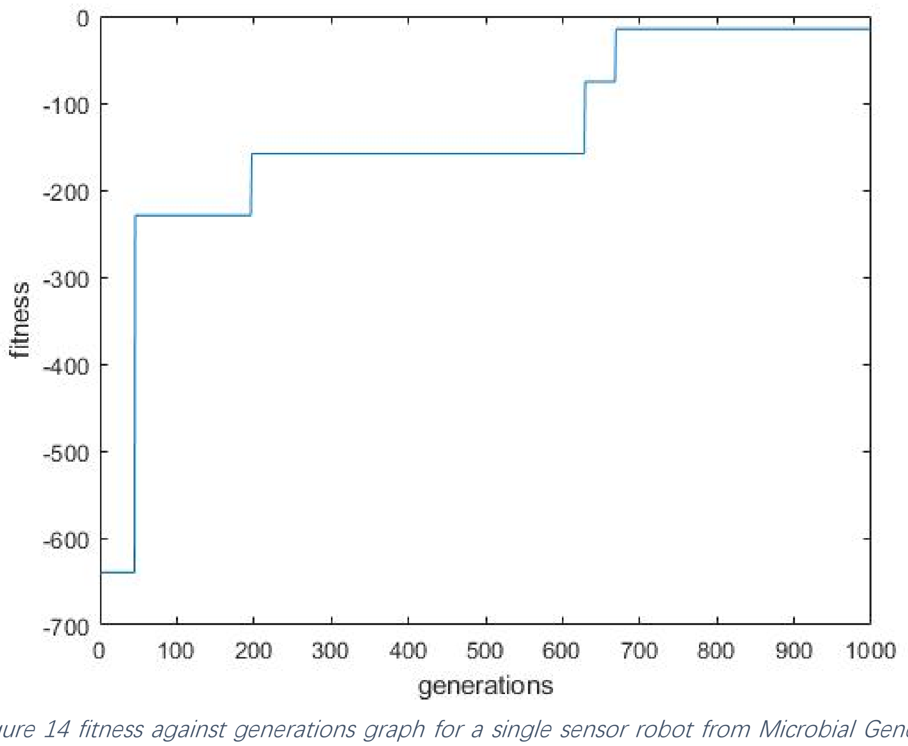

Figure 14 fitness against generations graph for a single sensor robot from Microbial Genetic Algorithm, it shows the solutions got better over the generations and eventually getting a very close location to the light.

3.7 Effect of the different starting locations

In the previous experiments in this study, the position was set to a constant (1,1) for the fairness of the comparisons. Here the study of how the starting position may affect the performance of the genetic algorithm robot.

Starting position (10, 10) is chosen as the comparison point because it is exactly 10 times the distance compared to all previous experiments. In figure 15, the fitness value for the robot at (10,10) having significantly worse fitness values but this is to be expected because the further starting

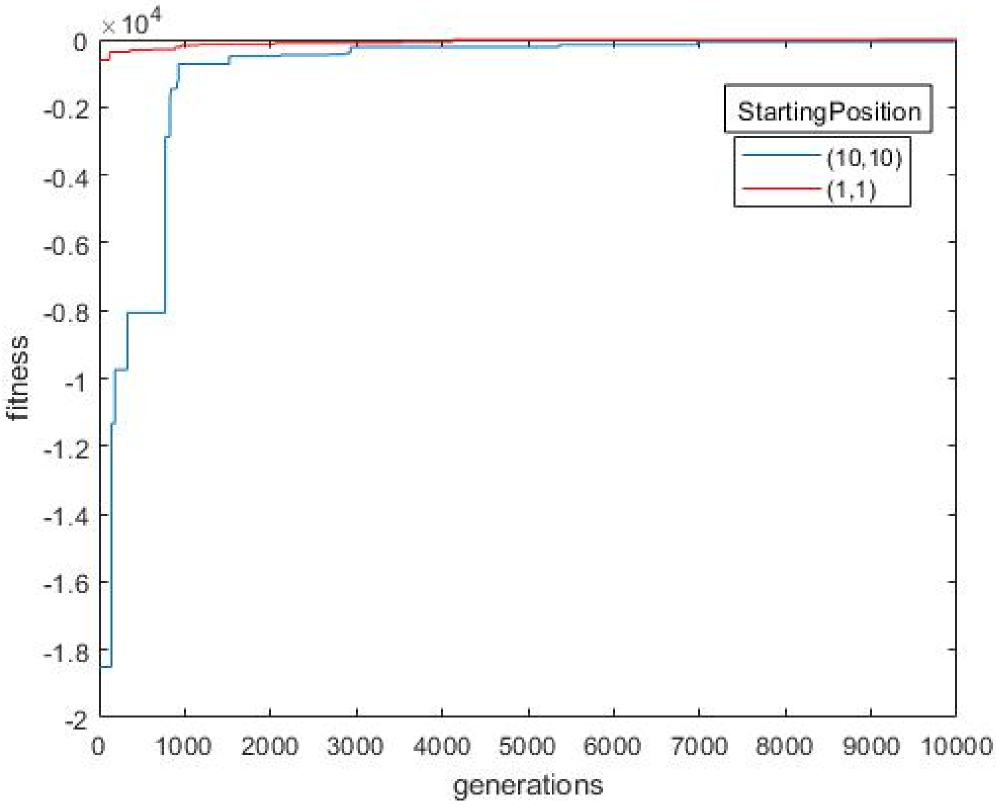

Figure 15 10,000generation fitness graph comparing the two different starting position of (1,1) and (10,10)

position means a lower fitness value. However, interestingly, the fitness value rapidly increases to relatively better fitness and eventually having almost identical fitness values as the robot with (1,1). This can be explained by the longer distance making the genetic algorithm find harder to obtain a suitable high fitness genotype than when it was at (1,1) and also it proves that after many generations, it is possible that the fitness value could be as good as that of the robot starting at (1,1). I decided to run it for 10,000 generations to allow the further away robot to converge to better genotypes.

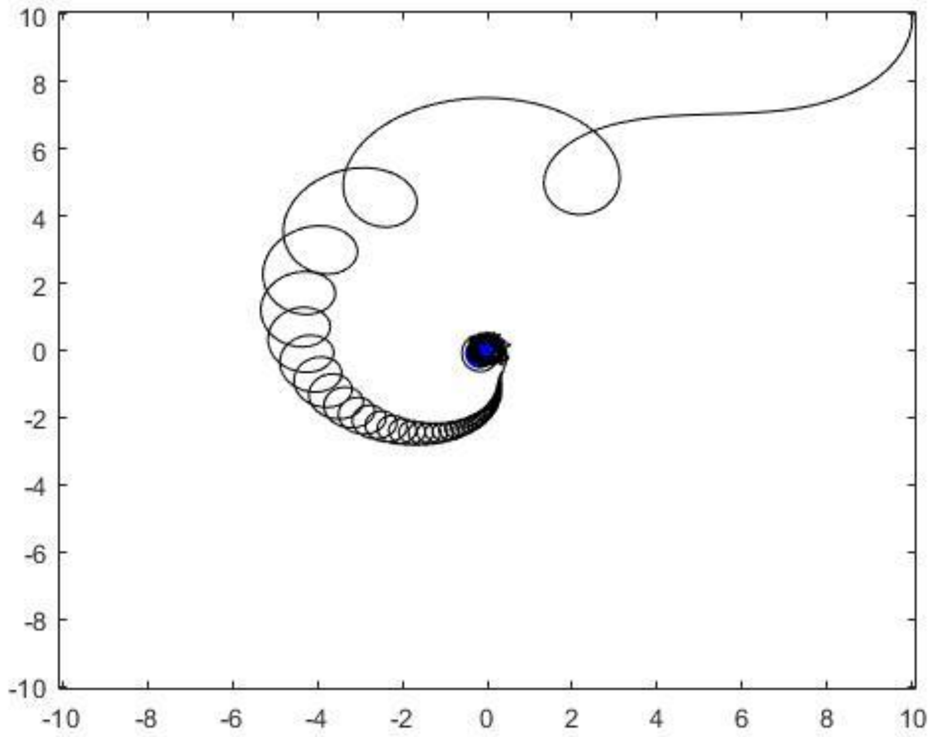

Figure 16 best performing robot trajectory with starting position at (10,10), the robot spirals initially slowly with bigger loops and gradually spirals smaller before reaching the light.

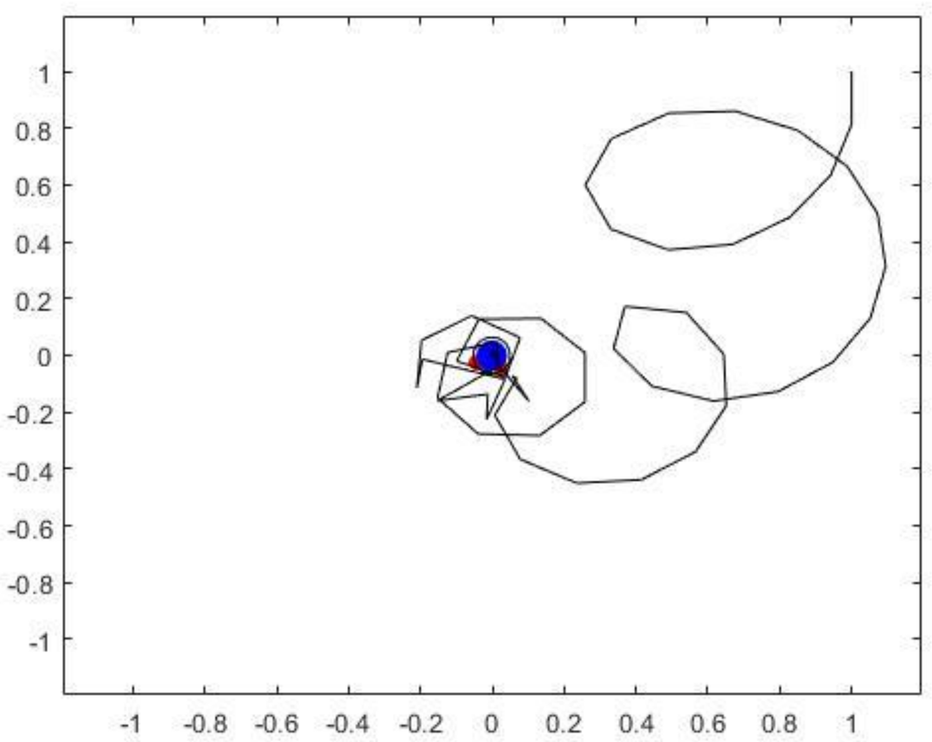

Figure 17 best performing robot trajectory with starting position at (1,1), shows the robot performing a similar spiral approach to reach the light and the robot in this figure is practically on the light and shows it will stay near the light even after overshooting.

4 Discussion

From the simulated robot experiments conducted in this study, I have been able to understand the concepts from Braitenberg (1986), where seemingly sophisticated behaviours could just be the result of some simple mechanisms. The simple behaviours of a single motor single sensor robot are showing the behaviour displayed by insects like cockroaches where if there is light it will run away.

Furthermore, bacterial chemotaxis in ecoli (DJ Webre, PM Wolanin 2003) is the example for these Braitenburg robots in action in the natural world. The flagellum propels the bacteria forward and causing the bacteria to make clockwise rotation(tumble) and counter-clockwise rotation to swim towards the chemical. It is like the spiral action from the robots when bias is added to the robot.

Inside our body, there are much contra-lateral connectivity and an example of it is that of the visual system. Whereby, the left eye is more strongly connected to the right hemisphere of the brain and vice versa. This could be indications that our ancestors were photophilic animals that go toward the light.

The genetic algorithm based simulated robots shows similarity with how animals adapt. Applying the microbial genetic algorithm, the evolution happens in the number of generations, the robot can iterate (1000 times in this case) and the reward in this problem is a distance away from the light source. Obviously, the fitter robots can get closer to the light source and pass their genes to the weaker robots. In this study, it has shown that the solution found by the genetic algorithm proves that a good genotype can be found to get the robot closer to the light source.

Similarities: The robots from figure 16, 17 illustrates that the best robots with different starting positions would still have similar behaviours and equally figure 13's single sensor robot also shares the same characteristics. It is very evident that the robot would all spiral towards the light source and that proves to be the best sort of approach to reaching the light.

Manual Vs GA generated robot

Both the manually designed robot and the genetically generated robot for the single sensor two motor robots shows similar approach in that they both spiral towards the light source but the manually designed robot does it with a bigger spiral making it less efficient at reaching the light source compared to that of the genetic algorithm generated robot.

0	0	-0.4632	0.7054	-2.2668	-2.5517	GA parameters
0	0	1	1.2	0	0.5	Manual designed parameters

The genetically generated robot can try many more values than manually designed algorithms, so the genetically generated robot can make use of many trial and error hence having values of many decimal places. In contrast, the bias for the GA is two negative values and that differed greatly to my own bias of 0 and 0.5.

In this study, I have proved my hypothesis that the genetic algorithm solution (parameters) are far better than the manually designed solution. And the starting position to have very little effect on the performance of the robot if given enough generations to find the most optimal parameters. And it is especially worth noting that some sophisticated behaviours observed are actually caused by very simple arrangements between the sensor and motor. It is interesting to apply biological inspirations to solve scientific problems as shown in this study, the perfect use of the biologically inspired microbial genetic algorithm that helps to find solutions for genotypes in order for the robot to get closer to the light.

5 References

Astonishing Hypothesis: The Scientific Search for the Soul, Francis Crick，1994

Simon, Herbert Alexander. The sciences of the artificial. MIT Press, 1969.

Braitenberg, Valentino. Vehicles: Experiments in synthetic psychology. MIT Press, 1986.

Harvey, I (2001) The Microbial Genetic Algorithm. Centre for Computational Neuroscience and Robotics, University of Sussex, Brighton

Eiben, A. E., et al. "The triangle of life: Evolving robots in real-time and real-space." Artificial Life Conference Proceedings 13. One Rogers Street, Cambridge, MA 02142-1209 USA journals-info@ mit. edu: MIT Press, 2013.

Brooks, Rodney Allen, and Luc Steels, eds. The artificial life route to artificial intelligence: building embodied, situated agents. L. Erlbaum Associates, 1995.