COURSEWORK 2, VEHICLES

Acquired Intelligence and Adaptive portfolio

JUNE 3, 2020

UNIVERSITY OF SUSSEX
Candidate no. 181334

Introduction

“In his 1984 book 'Vehicles: Experiments in Synthetic Psychology' Valentino Braitenberg proposes a number of hypothetical wheeled robots. He argues that even though the robots themselves were extremely simple, when embedded within a real environment, they would demonstrate quite complex, seemingly intelligent behaviours. Four of Braitenberg's Vehicles are of particular interest. Each has two ambient light sensors and two driven wheels, see Figure 1. The control architectures of these vehicles are considered 'representation free' in that sensors are connected to motors (at the wheels) by direct innervation which could either be excitatory (positive) or inhibitory (negative). In the case of negative connections, motors were considered to turn without stimulation (i.e. via baseline innervation, a so-called 'bias'). Beyond this, connections could either be contra-lateral (crossing, left sensor to right motor and vice-versa) or ipsilateral (same-side). The four possible vehicles using this architecture are thus: 1. Aggressor: Contra-lateral, positive connections. 2. Coward: Ipsilateral, positive connections. 3. Lover: Ipsilateral, negative connections 4. Explorer: Contra-lateral, negative connections.”[1]

We built each of the four Braitenberg robots and examined their behaviour using a simple simulation and plotted each trajectory along axis x and y.

1. *Aggressor: Contra-lateral, positive connections*
2. *Coward: Ipsilateral, positive connections*
3. *Lover: Ipsilateral, negative connections*
4. *Explorer: Contra-lateral, negative connections*

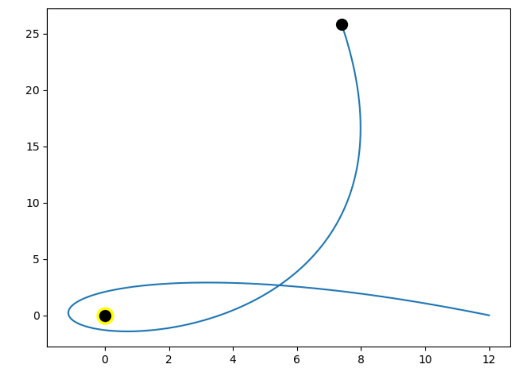

Figure 1 Aggressor trajectory plotted at (-12. 0). T = 500. Bearing = 150

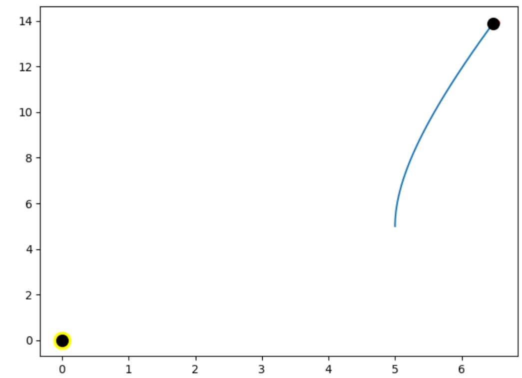

Figure 2 Coward trajectory Plotted at (5, 5), T = 100, Bearing = 90

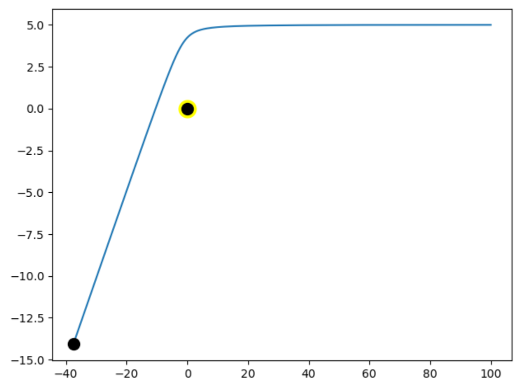

Figure 3 Lover trajectory Plotted at (100 , 5), T =150, bearing = 180

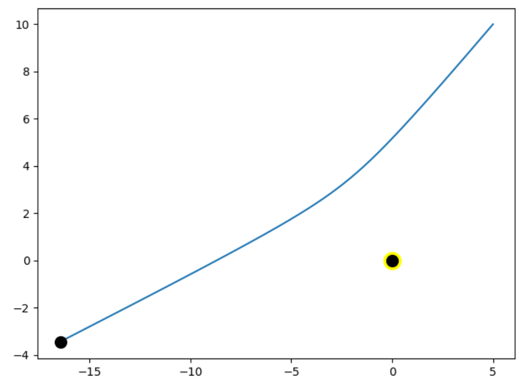

Figure 4 Explorer trajectory Plotted at (5, 10), T = 30, Bearing = 225

We then made a lesion on a Braitenberg robot such that we had a Braitenberg robot with only one sensor. We then implemented a controller that enables photo-taxis with only one sensor and recorded all observations for each vehicle.

We were able to create a lesion in the sample agent and created a controller to allow the one-eyed photo-taxis vehicle to reach the light source.

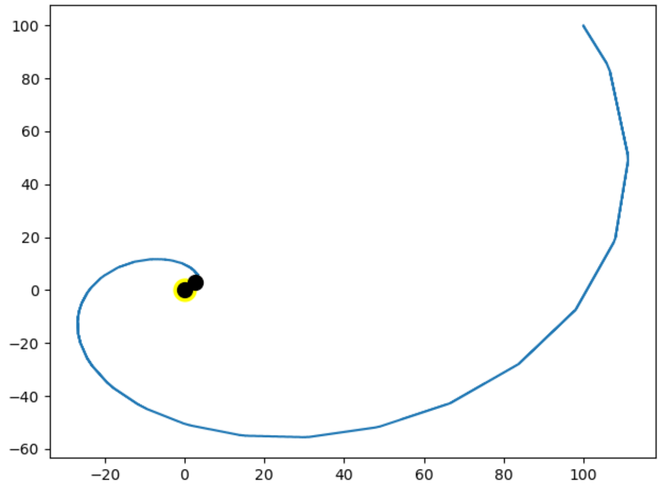

FIGURE 5 BRAITENBERG ROBOT WITH ONE LIGHT SENSOR.

Each of the above vehicles have been simulated to demonstrate their behaviour and, the extent to which the simple simulation paradigm allows for the behaviours predicted by Braitenberg" [3].

Using our one-eyed photo-taxis controller and making a lesion in the sample agent, we demonstrated the behaviour of the vehicle showing the path it took to get within 2 points of the light source where it stops moving from a variety of initial positions.

Method

Two-Sensor Braitenberg Models

To create each of the four Braitenberg models, we created a controller document. In this case we created the python scipt, vehicles.py in PyCharm IDE using python 3.8. [2]

In this script, we then import the 'run' method from the given Braitenberg agent, "simple_agent.py"[3].

"from simple_agent import run
import numpy as np"

We then create a function for each of the four Braitenberg models.

Each function needs to call the 'run'[3] method from 'simple_agent.py' and pass the parameters: "T, initial_pos, bearing, geno, plot_flag"[3].

1. *Aggressor: Contra-lateral, positive connections*

"Def Aggressor()

Geno = np.array([0, 1, 1, 0, 0, 0])

Initial_pos = [12, 0]

T = 500

bearing = 150

output = run(T, Initial_pos, bearing, Geno, 1)”

2. *Coward: Ipsilateral, positive connections*

Pseudocode:

“Def Coward()

Geno = np.array([1, 0, 0, 1, 0, 0])

Initial_pos = [5, 5]

T = 100

bearing = 90

output = run(T, Initial_pos, bearing, Geno, 1)”

3. *Lover: Ipsilateral, negative connections*

“Def Lover()

Geno = np.array([-1, 0, 0, -1, 1, 1])

Initial_pos = [100, 5]

T = 150

bearing = 180

output = run(T, Initial_pos, bearing, Geno, 1)”

4. *Explorer: Contra-lateral, negative connections*

“Def Explorer()

Geno = np.array([0, -1, -1, 0, 1, 1])

Initial_pos = [5, 10]

T = 30

bearing = 225

output = run(T, Initial_pos, bearing, Geno, 1)”

These methods can be called to run and display the trajectory graphs seen in figures 1, 2, 3 and 4.

One-Sensor Braitenberg model

We built and implemented a controller that enables photo-taxis with only one sensor. Firstly, we created another controller python script named “simple_agent_lesion.py” and copy paste the code from the original “simple_agent.py” and “vehiclesLesion.py”. In “vehiclesLesion.py” we run the following imports:

“from simple_agent_lesion import run
import numpy as np”

We then create the function in “vehiclesLesion.py”:

Pseudocode:

“Def navigateToLightSource():

    Geno = np.array([1, 1, 0, 0])

    Initial_pos = [45, -45]

    T = 100000

    Bearing = 5

    Output = run(T, Initial_pos, bearing, geno, 1)”

Now to create the lesion in the Braitenberg model, we need to remove the right sensor.

Then we made the following refactors to “simple_agent_lesion:

Comment out lines “34, 36, 79”

“w_rl = geno[2]” to “w_rl = geno[1]”

“bl = geno[4]” to “bl = geno[2]”

“br = geno[5]” to “br = geno[3]”

At line 57, added the code “iltemp = 0”

Replaced lines 81 and 82 with “lm = il*w_ll;” and “rm = il*w_rl;” respectively

From line 83, the lines of code added are:

“if(iltemp > il):
   rm = lm - 0.65;
if(iltemp < il):
   lm = rm - 0.65;

iltemp = il”

Once these vehicles have been created. We “implement each of the Braitenberg vehicles (as described above) to demonstrate both; their behaviour and, the extent to which the simple simulation paradigm allows for the behaviours predicted by Braitenberg.”[3]

Results

We have deployed each of the Braitenberg vehicles at a variety of X, Y coordinates and plotted the trajectories relative to the light. *(See intro for basic overview)*

##Direct all objects directly at light source and talk about results

Aggressor Vehicle Results

(See Appendix A)

Use of contra-lateral excitatory connectivity in the Braitenberg vehicle will cause the vehicle to turn towards the light source the closer it is to the light source when using two light sensors and 2 motors. The closer it gets to the light source, the faster it turns.

Coward Vehicle Results

(see Appendix B)

Use of ipsilateral connectivity in the Braitenberg vehicle will cause the vehicle to turn away from the light source and keep going in that direction.

Lover Vehicle Results

(*See Appendix C)*

Use of inhibitory ipsilateral connectivity with a bias keeping the motors on causes the vehicle to turn towards light sources. It appears, the closer the vehicle gets to the light source, the sharper it will turn. If It does not contact the light, it will keep going in the same direction after turning.

Explorer Vehicle Results

(See Appendix D)

Use of contra-lateral inhibitory connectivity, with bias, turns away from light sources and keeps going.

One-sensor Photo-taxi Vehicle Results

Discussion

We have successfully demonstrated how the braitsberg vehicles' paths are affected by light sources and turn in the predicted manner demonstrating behaviour from the following Braitsberg vehicles: contra-lateral excitatory connectivity, ipsilateral connectivity, Ipsilateral with inhibitory connectivity and a bias term keeping motors on, and contra-lateral inhibitory connectivity, with bias keeping motors on.

We have also successfully created a lesion in the simple Braitsberg agent (given in script "simple_agent.py") and modified the program in such a way in which it will always reach the light source stopping 2 pixels from the light source.

##Each of the above vehicles have been simulated to demonstrate their behaviour and, the extent to ##which the simple simulation paradigm allows for the behaviours predicted by Braitenberg."[3]

One limitation of this study is that it did not manage to demonstrate all possible behaviours of the Braitsberg vehicles as predicted by Braitsberg. Specifically, the plots of vehicle paths and tests carried out do not demonstrate the behaviour of speeding up and slowing down in both “Lover” and “Exploratory” vehicles. Although this study is not direct evidence of this phenomenon occurring, the observations of the amount of turning and the severity of the turning circle, could be indications that the vehicles had either sped up or slowed down.

A second limitation of this study, the simple agent with legion and controller is very time inefficient from certain initial positions as it has the tendency to head off in a straight line for an uncertain amount of time before turning and directing towards the light. In every instance, the vehicle reached 2 points from the light source.

such as detecting the simple simulated vehicles and vehicle path plots simulation paradigm

Conclusion

We have investigated and exemplified a variety of key behaviours predicted by Braitsberg.

A combination of the choice of model and methodology used to conduct this experiment meant that the key behaviour of speed was not able to be observed. This means that we were unable to extensively prove all possible behaviours. However, although this means that 2 of the 6 key behaviours were not able to be measured observed, we can still observe a potential side effect of the phenomenon occurring in the form of an altered vehicle path turning circle.

Further investigation would be needed to conclude whether the altered turning circle severity has a correlation with the vehicle speeding up or slowing down.

A further study would primarily investigate the key features of speed in relation in both explorer and lover vehicles. Doing this would allow us to conclude the objective of finding out if all the Braitsberg vehicles predicted characteristics are present. As currently, this simulation and experimental design is limited in that it only allows us to study the turning direction for each combination of connectivity and bias.

We observe that the vehicle using contra-lateral excitatory connectivity with no bias behaves differently from the vehicle using contra-lateral excitatory connectivity with bias. We can also observe that the vehicle using ipsilateral connectivity with no bias behaves differently from the vehicle using ipsilateral connectivity It can be deduced from our data that there could be a possible correlation between the addition of the bias and the severity of the turn observed in vehicle path away from or towards the light source.

Further investigation could be carried out to fully document the effects of bias on both contra-lateral excitatory connectivity and ipsilateral excitatory connectivity in Braitsberg vehicles.

Further investigation is *needed* in investigating the

Future Work and Studies:

Investigate, time spent slowed down, ratio of full speed(motors at 1) to motors inhibited, with and without bias on both lover and exploratory vehicles.

Investigate modelling a more efficient model to reach the light. The one sensor requires a very high amount of time to reach the sensor.

References

[1] Buckley, C., 2020. *CW2 Vehicles*. [ONLINE] Available at: https://canvas.sussex.ac.uk/courses/8727/files/1254773?module_item_id=705866. [Accessed 01 June 2020].

[2] JetBrains. 2020. PyCharm: the Python IDE for Professional Developers by JetBrains. [ONLINE] Available at: https://www.jetbrains.com/pycharm/. [Accessed 01 June 2020].

[3] Buckley, C., 2020. *CW2 Vehicles.* [ONLINE] Available at: https://canvas.sussex.ac.uk/courses/8727/files/1106078?module_item_id=653796. [Accessed 01 June 2020].

Appendix

Appendix A

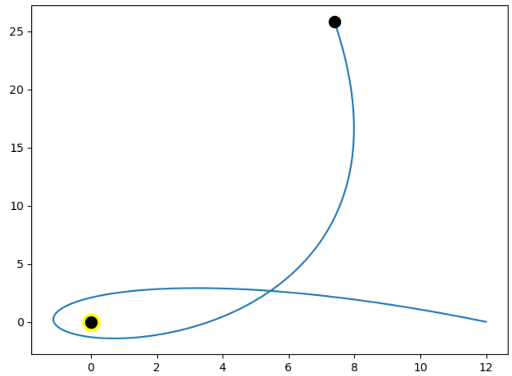

FIGURE 1 PLOTTED AT (-12, 0), T = 500, BEARING = 150

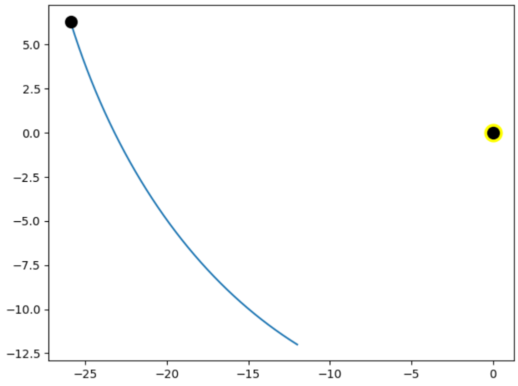

FIGURE 2 PLOTTED AT (-12, -12), T = 500, BEARING = 150

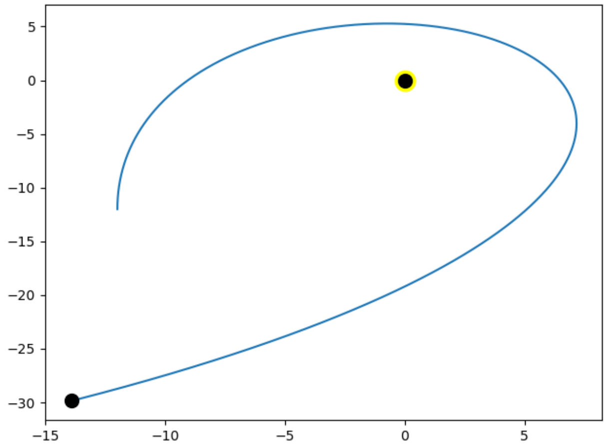

FIGURE 3 PLOTTED AT (-12, -12), T = 1000, BEARING = 90

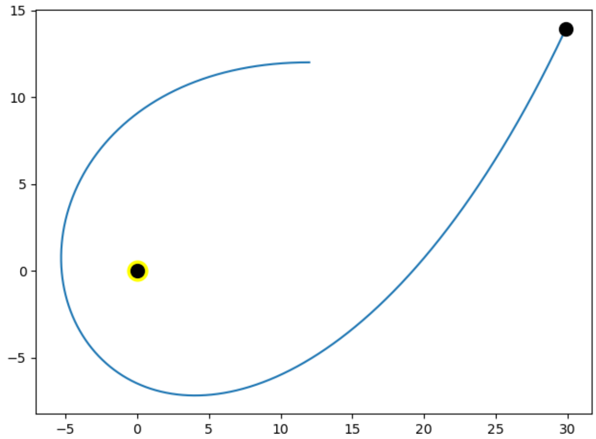

FIGURE 4 PLOTTED AT [12, 12], T = 1000, BEARING = 180

Appendix B

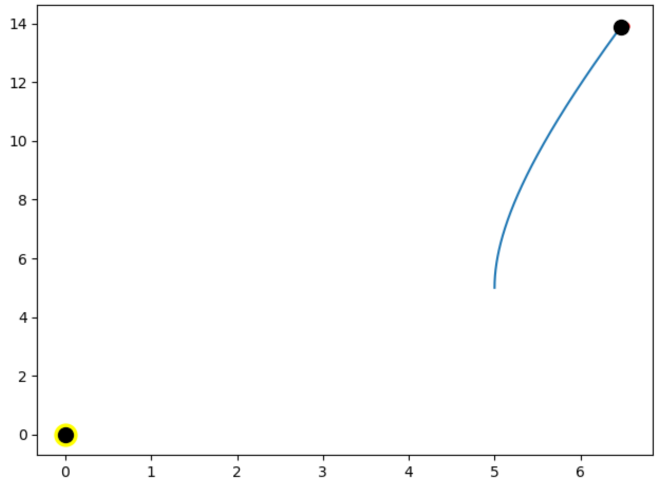

FIGURE 5 PLOTTED AT(5, 5), T = 100, BEARING = 90

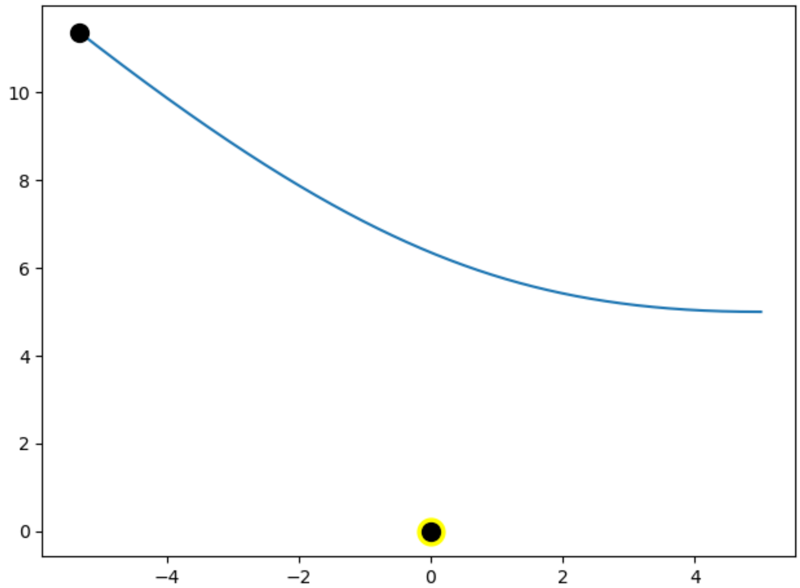

FIGURE 6 PLOTTED AT (5, 5), T = 100, BEARING = 180

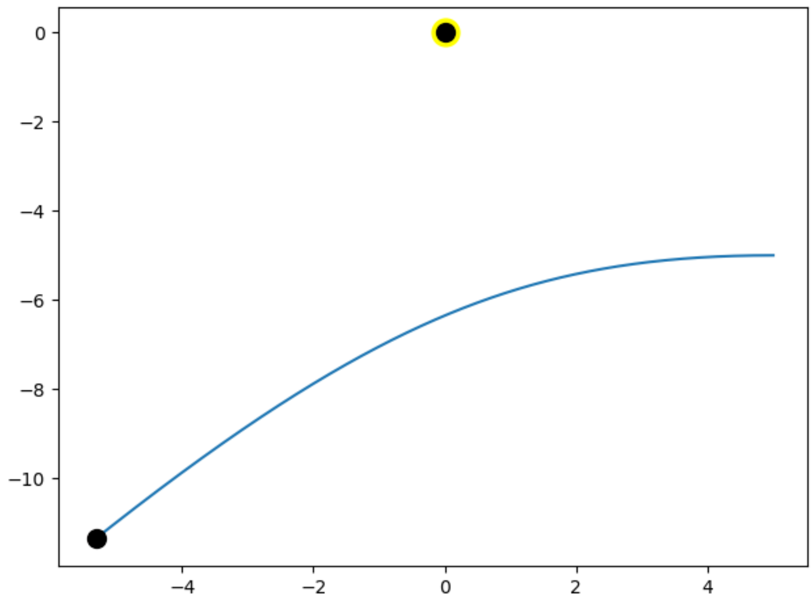

Figure 7 Plotted at (5, -5), T = 100, Bearing = 180

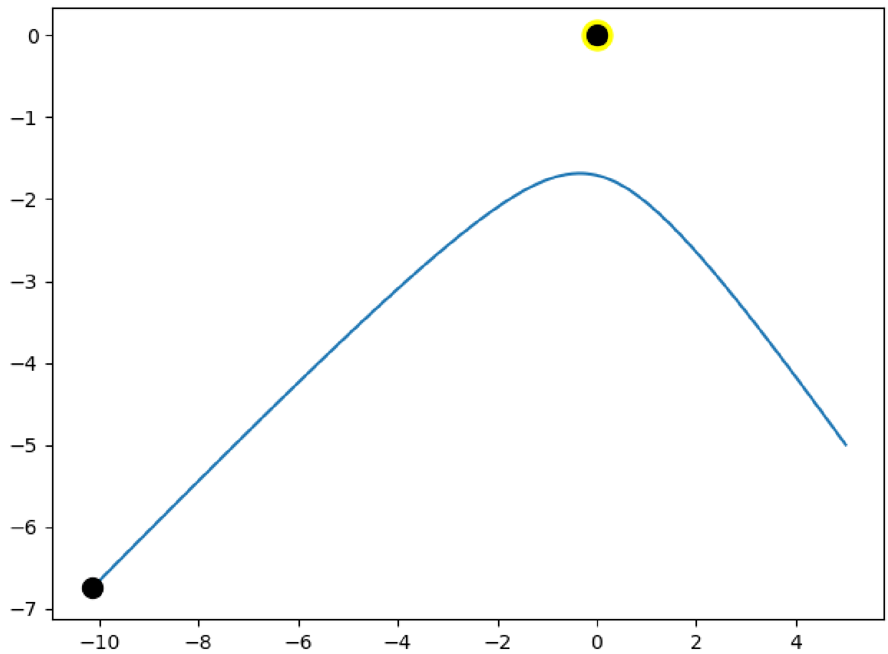

Figure 8 Plotted at (5, -5), T = 100, Bearing = 140

Appendix C

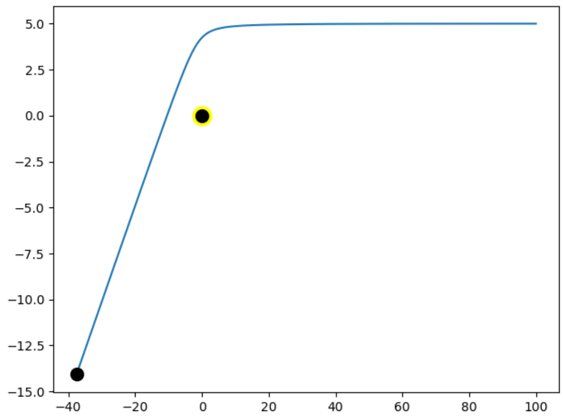

FIGURE 9 PLOTTED AT (100 , 5), T =150, BEARING = 180

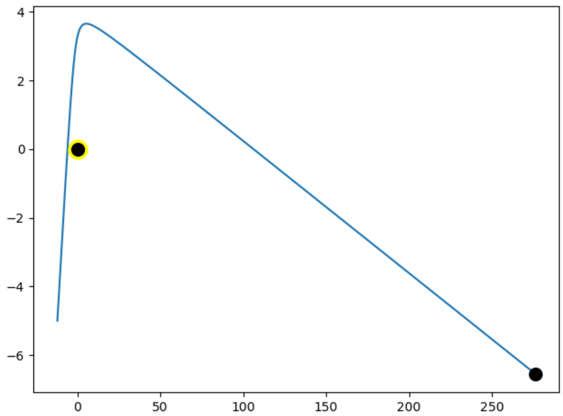

FIGURE 10 PLOTTED AT (-12, -5), T = 300, BEARING = 40

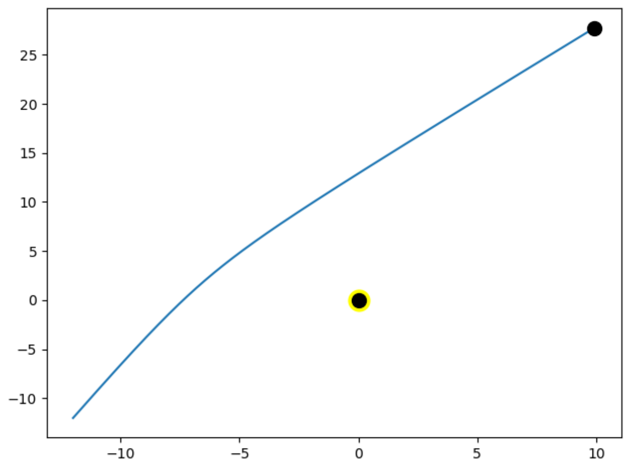

FIGURE 11 PLOTTED AT (-12, -12), T = 50, BEARING = 70

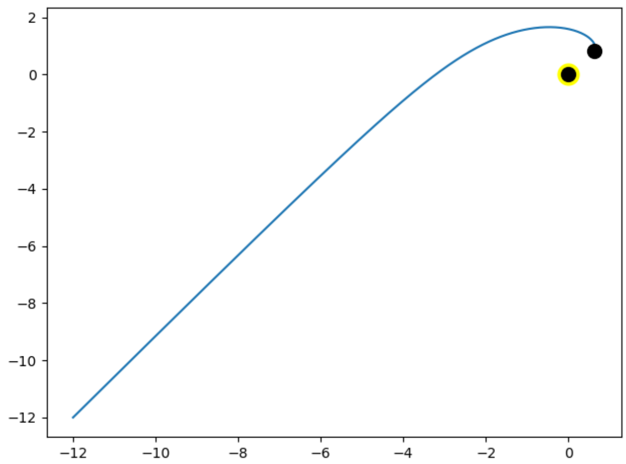

FIGURE 12 PLOTTED AT (-12, -12), T = 500, BEARING = 55

Appendix D

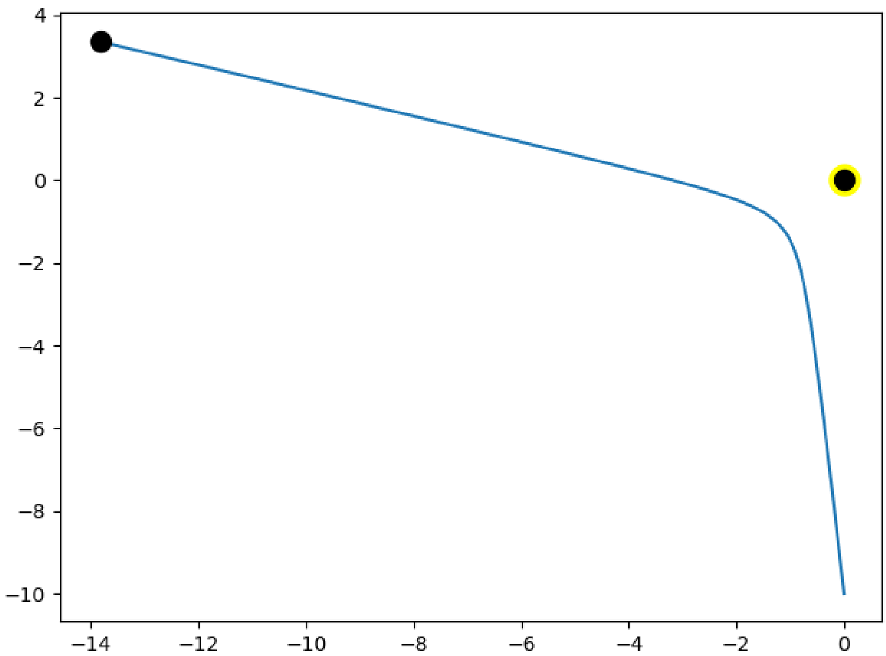

FIGURE 13 PLOTTED AT (0, -10), T - 30, BEARING = 95

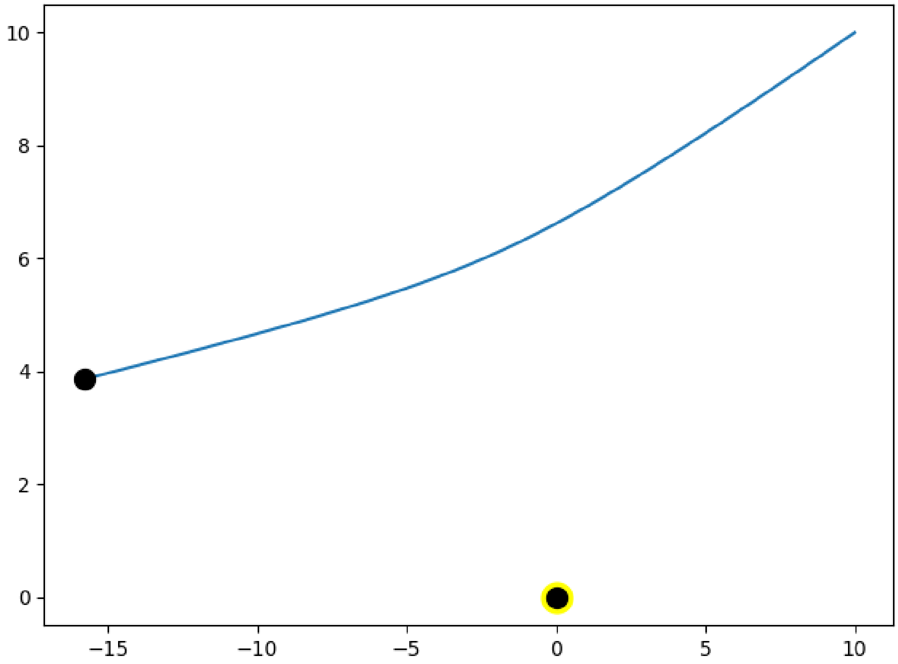

FIGURE 14 PLOTTED AT (10, 10), T = 30, BEARING = 200

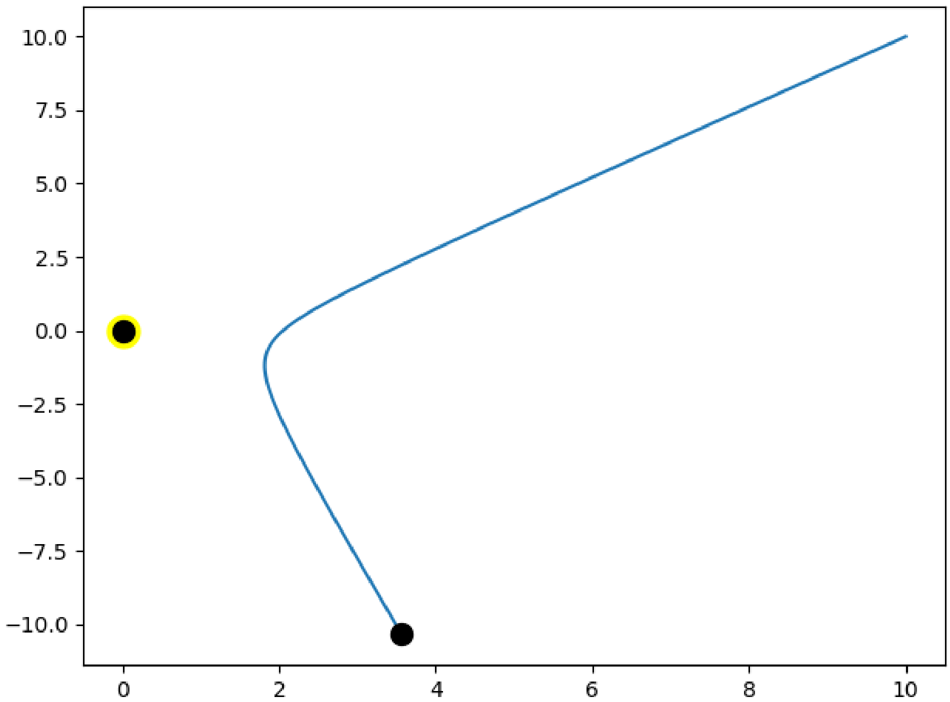

FIGURE 15 PLOTTED AT (10, 10), T = 30, BEARING = 230

Appendix E

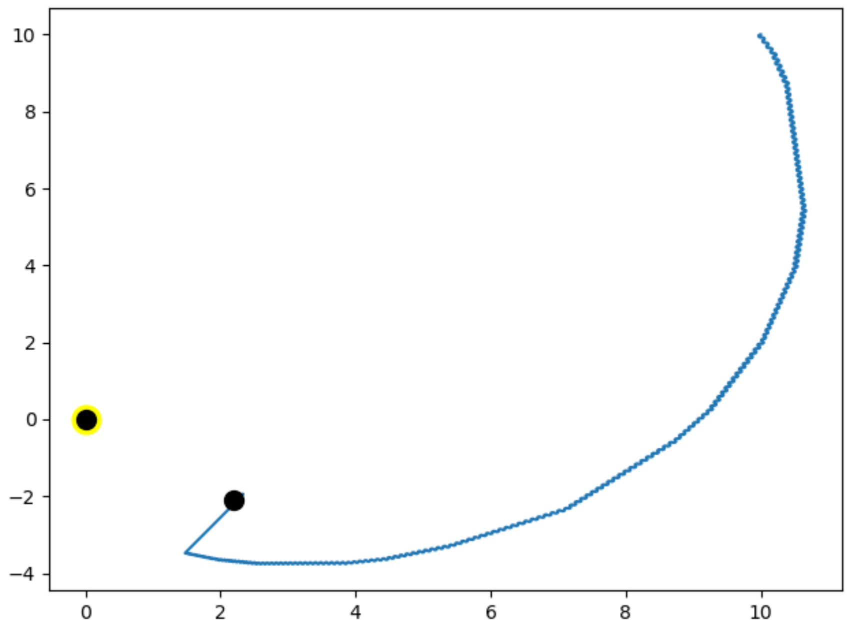

FIGURE 16 PLOTTED AT (10, 10), T = 1000, BEARING = 20

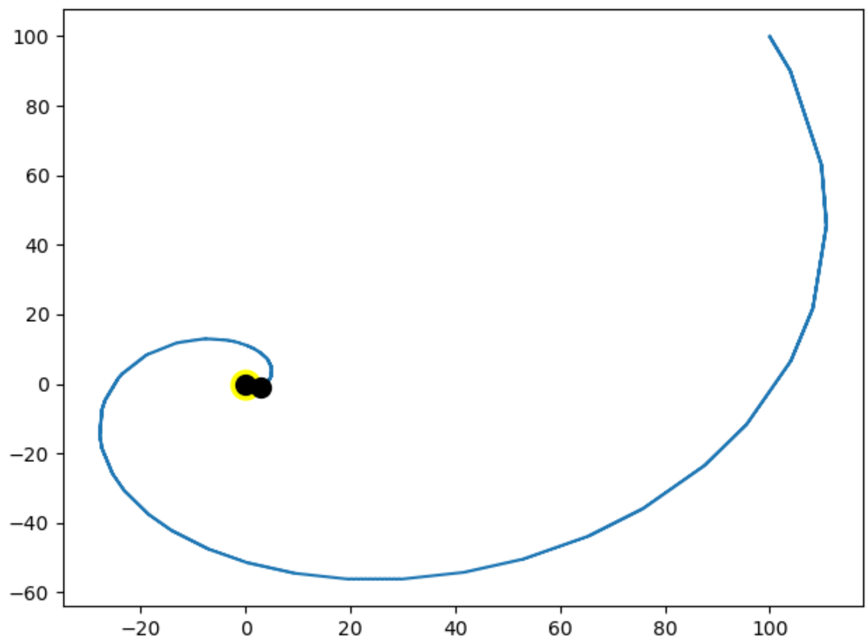

FIGURE 17 PLOTTED AT (100, 100), T = 10000, BEARING = 0

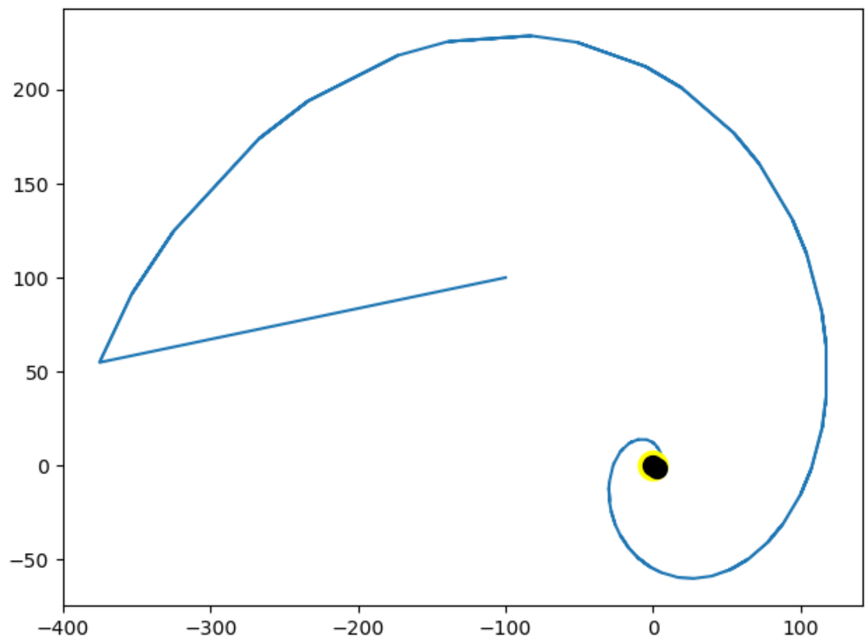

FIGURE 18 PLOTTED AT (-100, 100), T = 10000, BEARING = 0

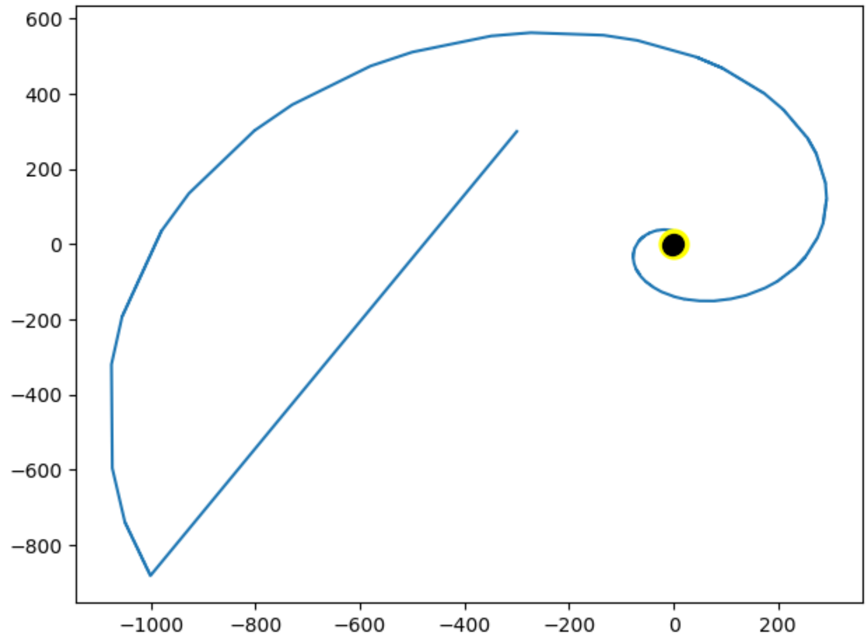

FIGURE 19 PLOTTED AT (-300, 300), T = 20000, BEARING = 50

References

[1] Buckley, C., 2020. *CW2 Vehicles*. [ONLINE] Available at: https://canvas.sussex.ac.uk/courses/8727/files/1254773?module_item_id=705866. [Accessed 01 June 2020].

[2] JetBrains. 2020. PyCharm: the Python IDE for Professional Developers by JetBrains. [ONLINE] Available at: https://www.jetbrains.com/pycharm/. [Accessed 01 June 2020].

[3] Buckley, C., 2020. *CW2 Vehicles.* [ONLINE] Available at: https://canvas.sussex.ac.uk/courses/8727/files/1106078?module_item_id=653796. [Accessed 01 June 2020].