Interim Report
Final Year Project
2020 – 2021

Xiang Mao

20251952

Barak Pearlmutter

Learning Chemotactic Strategies

table of Contents

1. Goals of this project

1.1 Final Goal

a) augment the simulator to include additional effects (turbulence, particle decay, 3D)
b) augment the agent to allow more sophisticated strategies.
c) attempt to learn chemotactic strategies appropriate at various length scales.

1.2 Stage goals

a) Run the basic strategy
b) Collect data and analyze the impact of parameter changes

2. Problems encountered and progress so far

2.1 Problems and solutions

2.1.1 Lack of program runtime environment

Problems

Even though I have installed python and some plugins on my windows system, I still encounter problems in the IDE.

Solutions

I used PyCharm and used it to manage my python plugins, and the problem was solved.

2.1.1 The code extensibility

Problems

The initial code was a single file with many functions concentrated in the main() function, which would have taken more time to modify the program code.

Solutions

I put the agent as a class in another file, so that when modifying the agent or the main function, it will not affect each other. The operation is very clear.

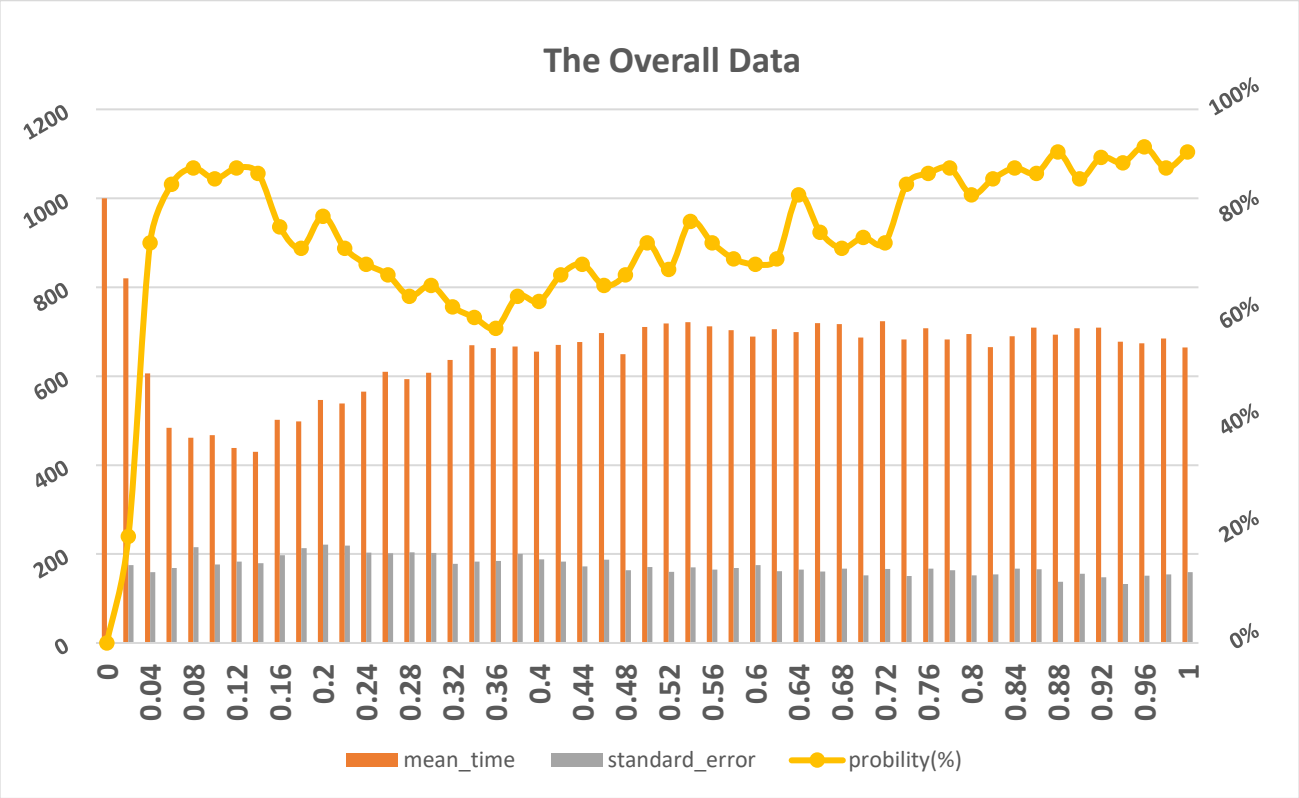
2.1.2 Basis for improving parameters or algorithms

Problems

There is no basis for improving the parameters or algorithms, and it is difficult to determine what level of modification to the parameters is optimal.

Solutions

With over 5000 runs, some samples were obtained as a basis for optimizing the parameters. Plot (1-1) shows a chart of the overall results of the agent runs drawn from the sample data. This will be one of the general bases for the optimization of the parameters.



(1-1)

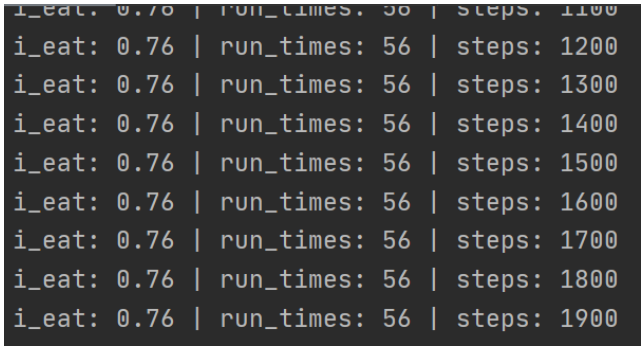
2.1.3 Ensure that the program runs continuously

Problems

When I use python's installation library to generate an executable to run a file, the program always stops working within a few seconds of startup. There is no way to get to the status of the current program run until it is fully run.

Solutions

by adding a step variable, and output it every 100 steps, as well as outputting the number of eat parameters and the total number of runs for each completed. As (2-1)



2-1

2.2 progress so far

2.2.1 Get the data for optimization

A lot of data are obtained by running the program, and these data are able to optimize the current parameters

2.2.2 Storage Data

The data can now be stored in files and the results of each run will be stored in a file with a different name. The storage operation is simpler than using Mysql.
I have tried to use Mysql database to store data, but it is very tedious to install and configure. There are a lot of sql statements to be written to process the data, but at the moment only the basic data needs to be obtained and it takes a lot of time to do so. If I don't continue to use Mysql to process data after I get a lot of data later, it won't be worth it.

2.2.3 Data processing

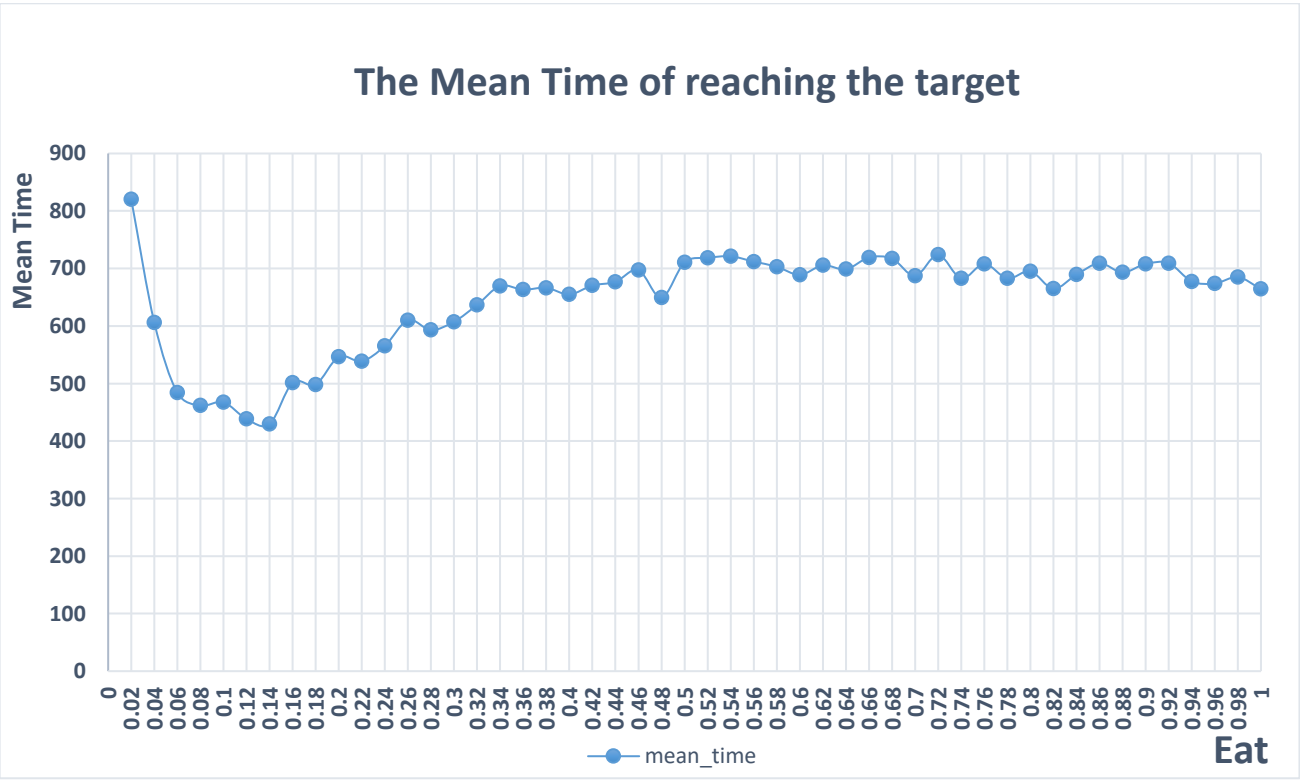
At present, I can use the DataBox class to calculate the data, or directly put the data into Excel, and use the pivot table to process the data.

2.2.4 Analyze and display the data

Currently I can use the matplotlib library to present the data and also use the graphs in Excel to present the data.
By (3-1) we can know that the agent cannot reach the source at parameter eat=0. In the interval from 0.04 to 1 and with eat=0.36, the agent has the lowest success rate of reaching the source with a probability of P=59%.
The probability is highest for eat=0.96, with a probability of P=93%. This will be one of the general bases for optimization of the parameters.

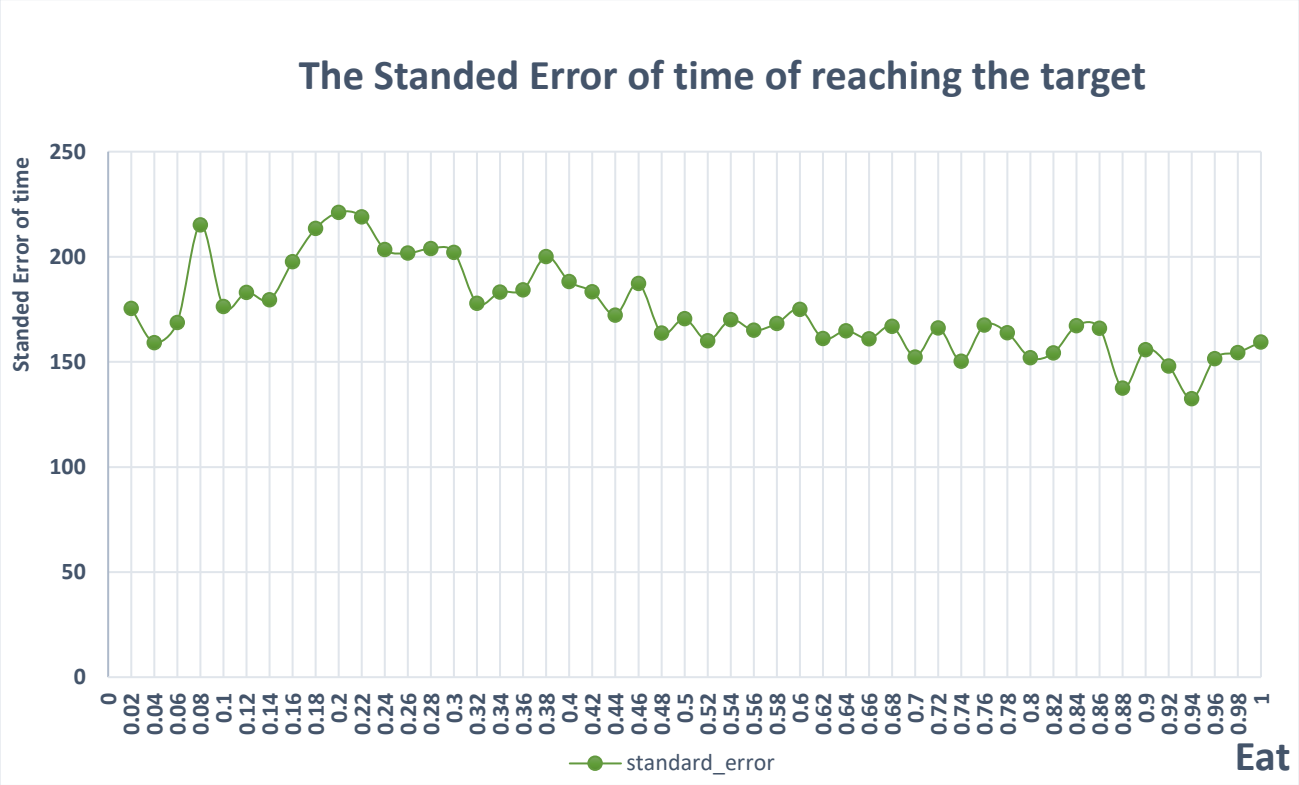
(3-1)

Plot (3-2) shows the mean time of the time taken by the agents that successfully reach the source. From this we can see that at eat=0.14, the agent consumes the shortest time to reach the source. This will be one of the general bases for optimization of the parameters.



(3-2)

Plot (3-3) shows the standard deviation of the time taken by the agent to successfully reach the source. From this we can know that the agent's running time is most stable at eat=0.94. This will be one of the general bases for optimization of the parameters.



(3-3)

2.2.5 Optimize the code

Processing the data by a DataBox class and a Data class makes coding much easier, although the improvement in runtime is not significant. Optimizing the program will make the rest of the work much smoother.

3. Next steps

3.1 Adjust parameters based on data

Adjusts the parameters of the agent based on the data obtained so far, thus making the agent smarter.

3.2 New parameters and new data

Run the agent with the new parameters adjusted and try to modify the other parameters to get the new data.

3.3 Modify the algorithm

After tuning the parameters, data are obtained using the new reinforcement learning algorithm and comparing what differences there are in the optimal results that can be achieved by the different algorithms.

3.4 Enables programs to optimize themselves

Try to make the program work better by being able to modify the parameters based on the results of a large number of runs.

4. General comments

4.1 Run faster

4.1.1 Matplotlib

After I turned off the real-time drawing image function using python matplotlib library, the running speed was significantly improved.
The use of Matplotlib has a great impact on efficiency and is not suitable for use in the process of data collection, but because it has very rich chart content, it is more convenient for data display.

4.1.2 Pygame

After closing the pygame form, the running speed increased slightly.
Although closing pygame does not significantly increase the running speed, there is no need to display the pygame window in the process of collecting data by repeatedly running the program.

4.1.3 Running time

More than 5000 runs took about 4 days in total.
In the future, it may try to improve efficiency based on the optimization of the algorithm.
Or use some libraries that improve the running speed, such as numba.

4.2 Reinforcement learning library

Python has a very rich library, including many machine learning libraries (for example: sklearn), I may be able to try these libraries before using new algorithms.

4.3 Data analysis library

Maybe using some libraries for data analysis will give more information.