Final Report

Team Name: Sun Knight team Leader: 杨晨宇 team member:韩冰

Date: 2017-1-2

**1 Prototype System Introduction**

* 1. **Functions**

1.experimental functions: The system is able to conduct an experiment with demon algorithm, and return the experimental result back in the form of graphs. One plot graph shows the relationship between N and energy, and other two graphs describe the demon energy in two ways: by analyzing the distribution of demon energy and by show the energy data corresponding to each steps. And a graph shows the distribution of the energy of particle after finishing the last step.

2.assistant functions: The system offers user-friendly GUI and some assistant functions. There is a line to show the current progress. And there is input check function to give feedback warnings if the users input an illegal number. A textbox is placed besides the graphs to show the specific data. Two scales is set for users to change the number of columns. And the system offers functions to save the data and the graph. And users can change the back color if they like.

3.3D-visual functions: The system can show 3D windows to give users a more intuitionistic knowledge about how particles collision with each other.

**1.2 Running Environment**

Windows 10

**Developing Environment**

Python 3.4.4

Vpython

matplotlib

PyScripter 2.5.3

Pycharm

…

**2 Task Allocation**

The tasks for each team member.

杨晨宇: demon algorithm and data collection and transformation into graphs.

A 3D program using pridiction algorithm.

韩冰：GUI and assistant functions associated with it and 3D program using collision algorithm

**3 System Architecture**

**3.1 User Interface Component**

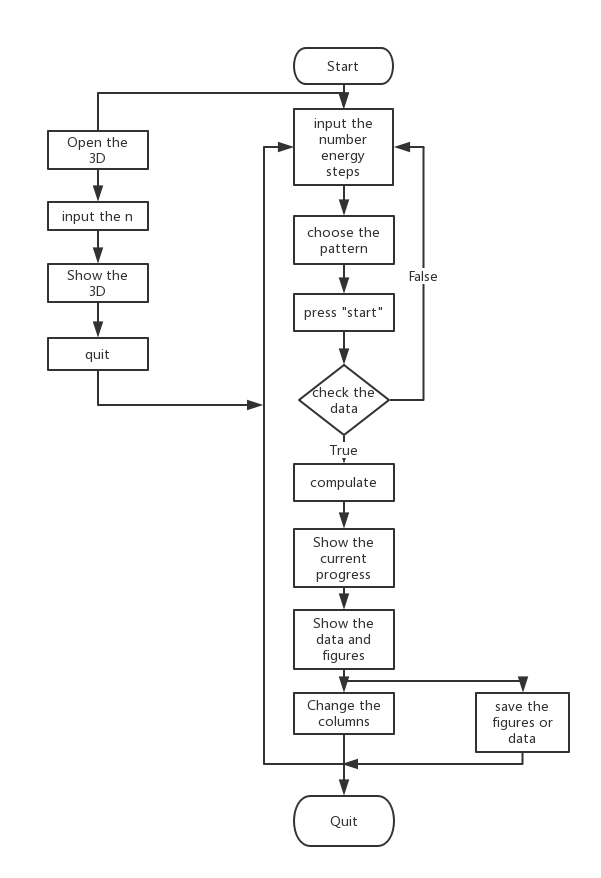
User interface contains the main function which is input and output, and other assistant functions like change column number, input checking and data save and current progress feedback.

**3.2** **Simulation and Visualization Component**

Simulation and visualization contains 3 layers of data processing, first is simulation and then collect the data that is needed. And at last call matplotlib to draw 4 pictures according to the raw data.

**4 Algorithm Description**

**4.1 User Interface Component**

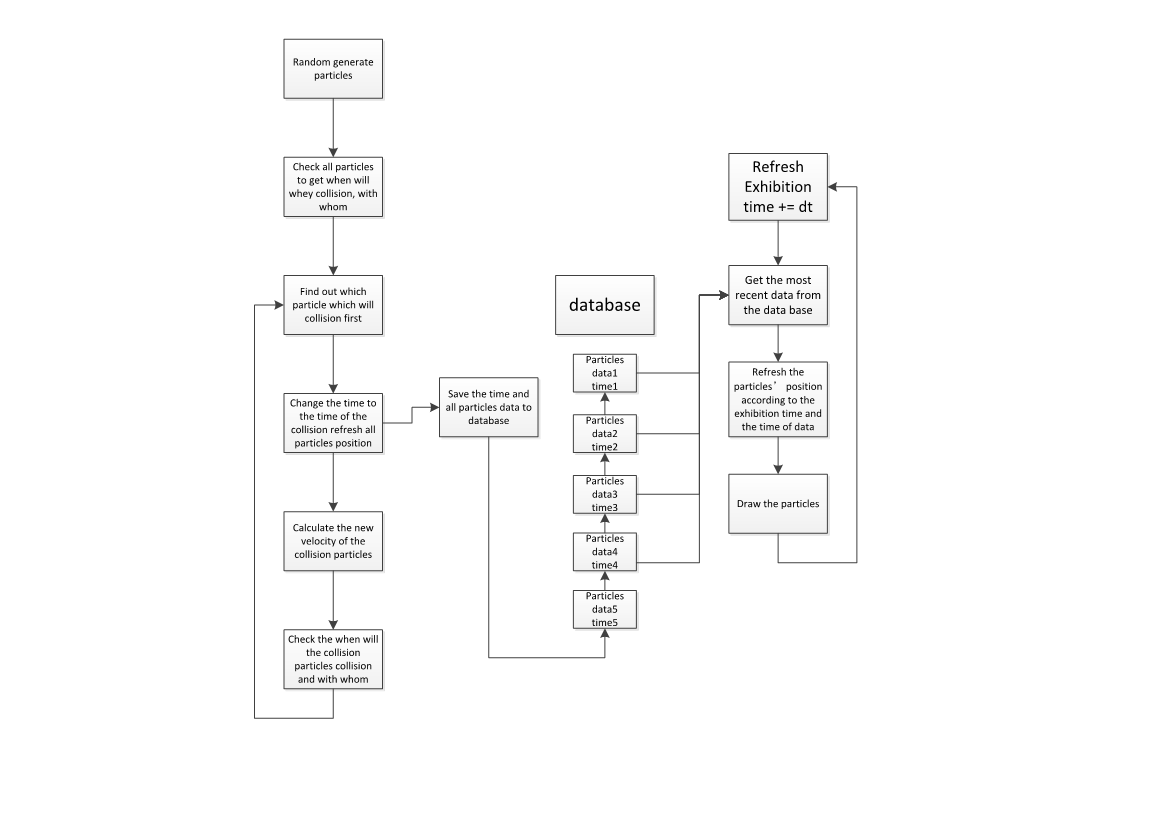


**4.2 Simulation Component**

Flowchart of the collision algorithm

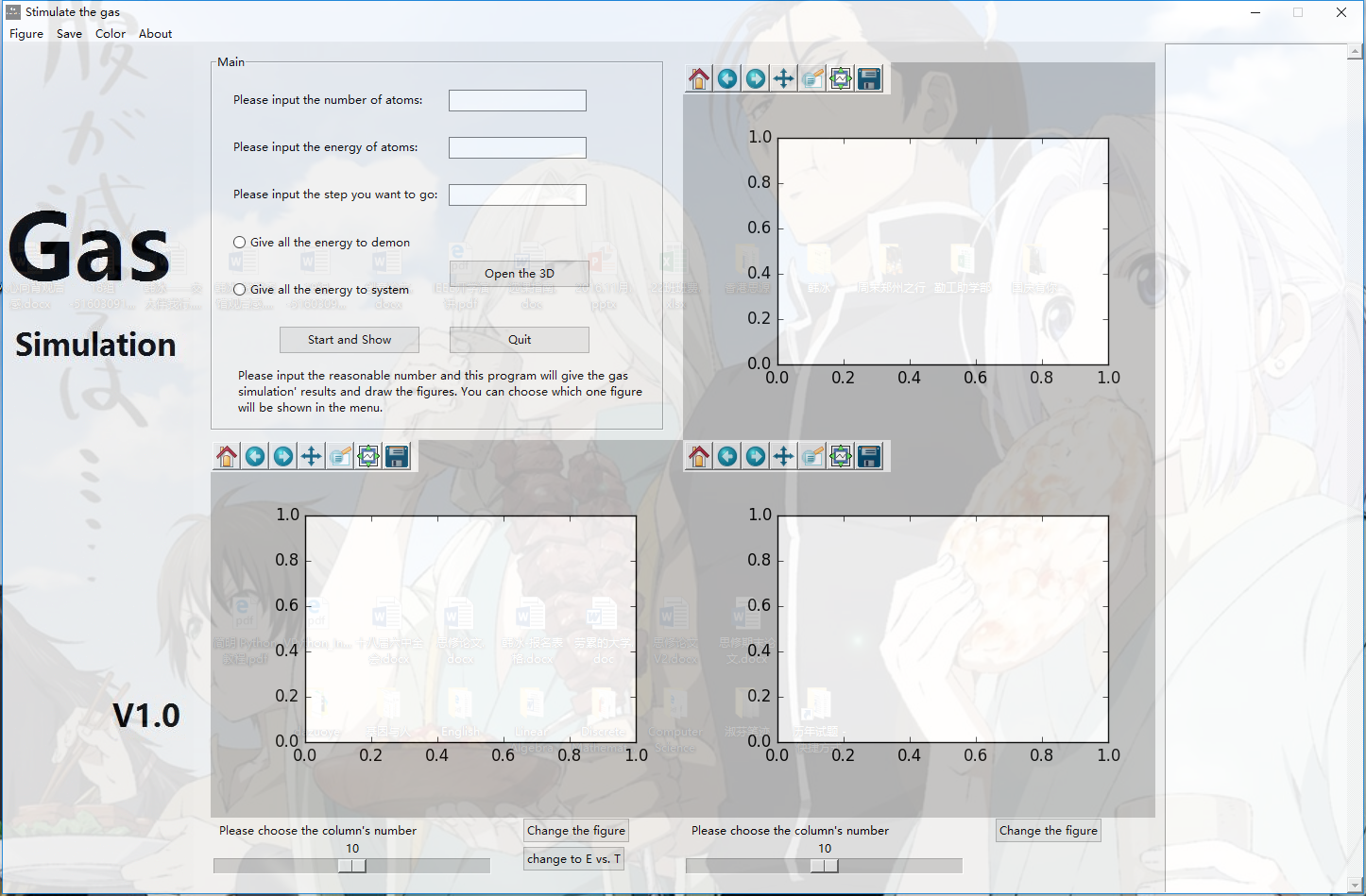


Flowchart of the prediction algorithm

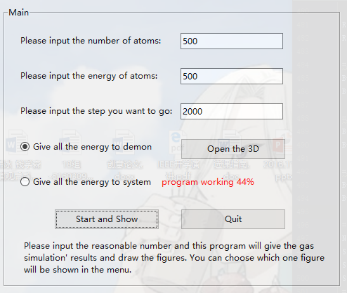
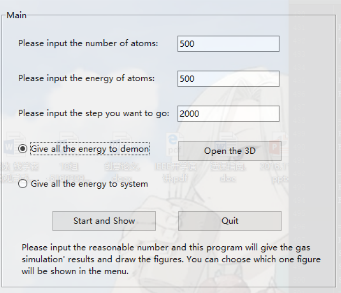


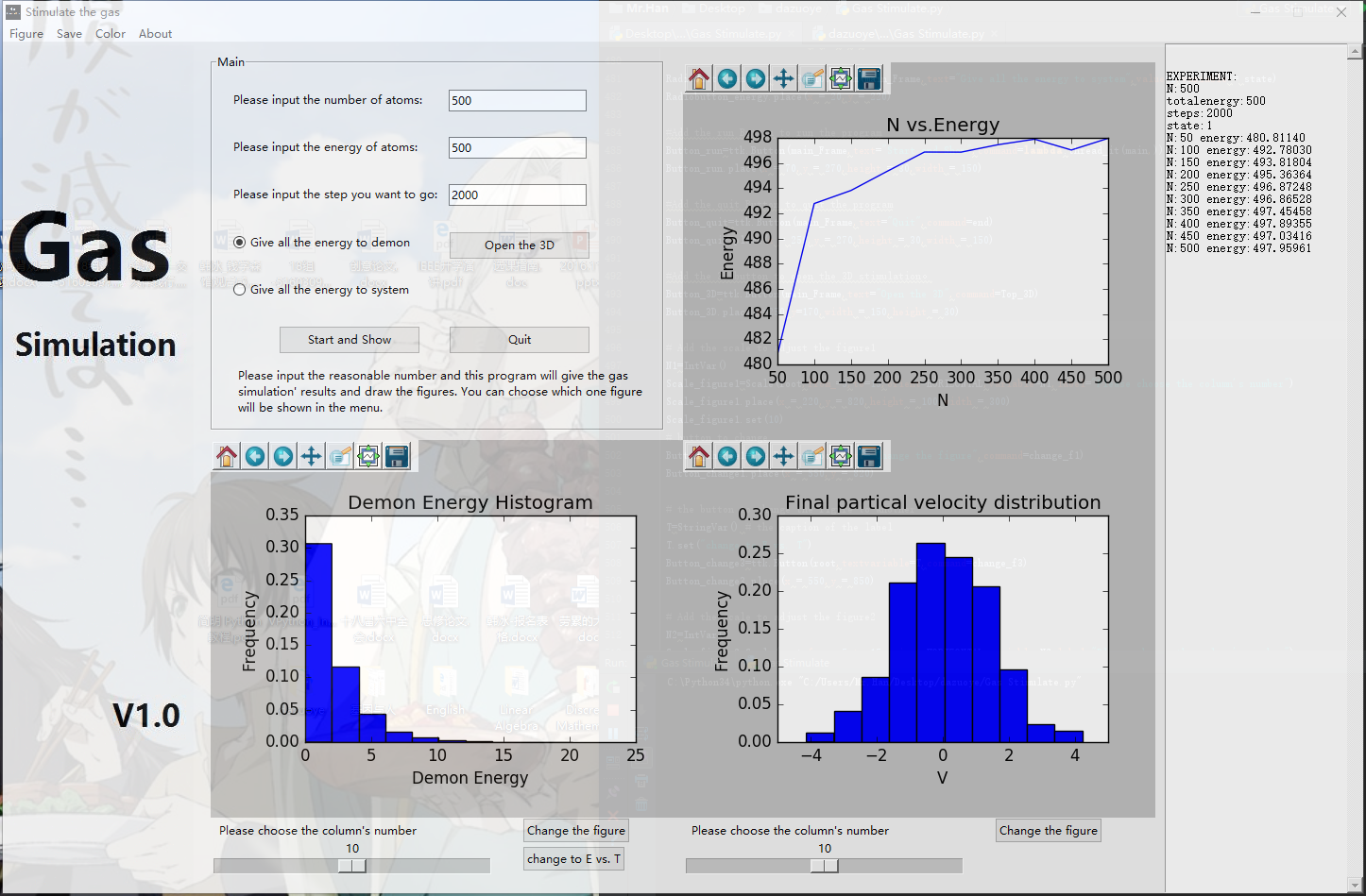
**5 Demo and Testing Result**

**5.1 Screenshots**

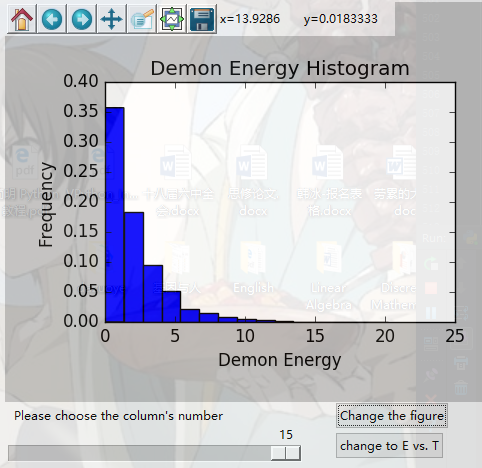
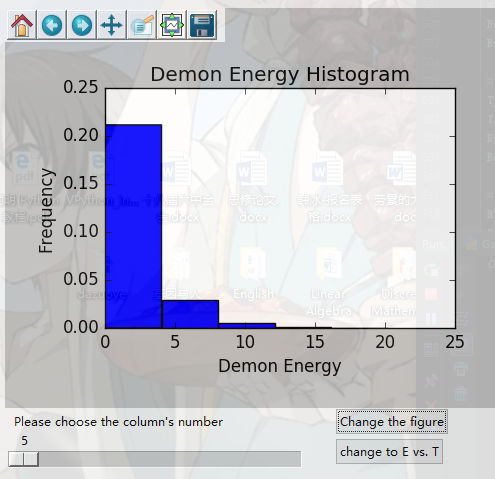


User interface

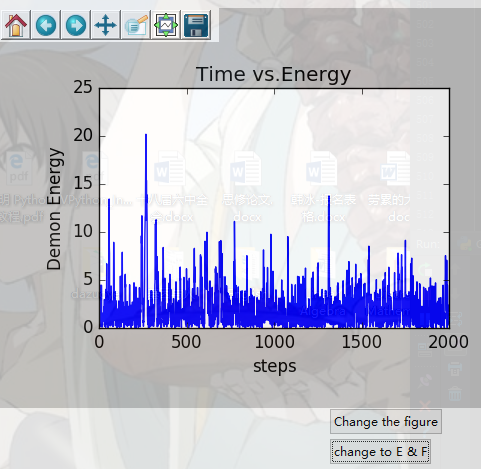


Input input-check and current progress

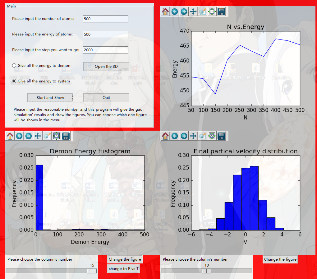
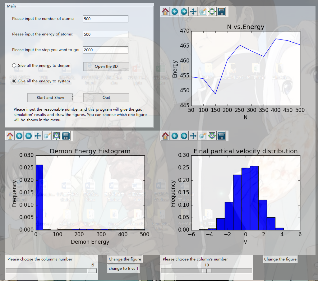
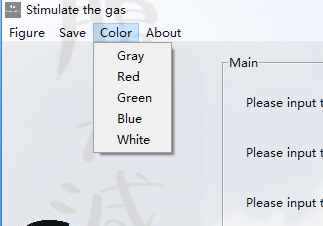
Result output



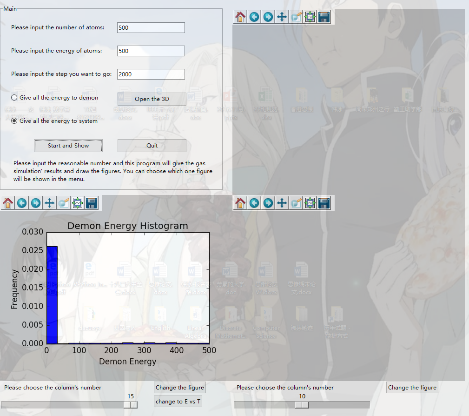
Change the number of columns



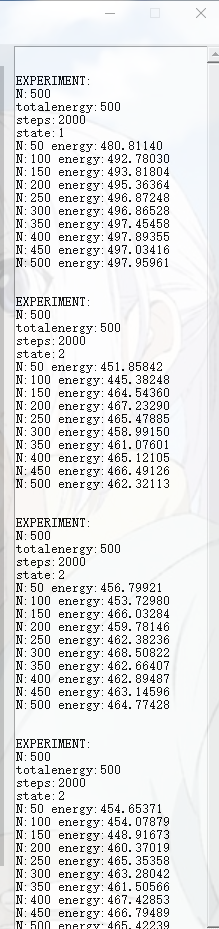
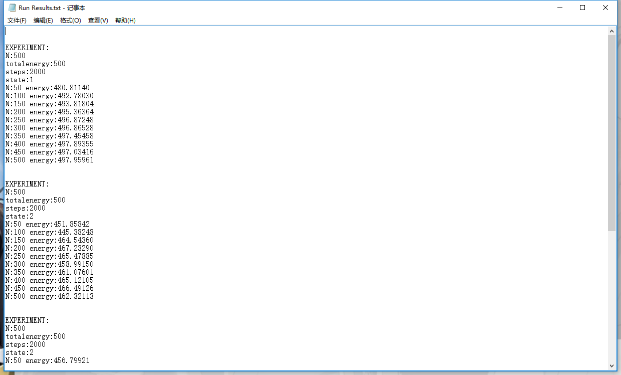
The graph of demon-energy-time



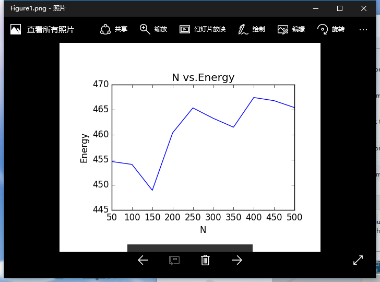
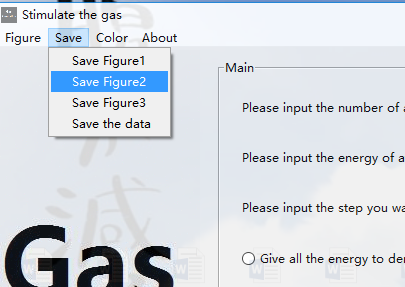
Change the background color



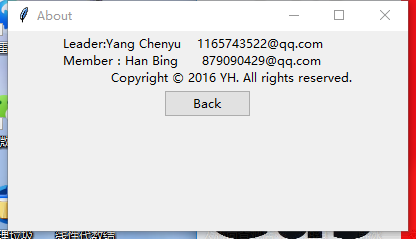
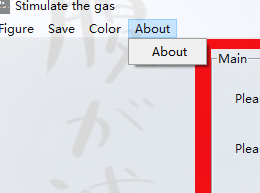
Hide the graphs



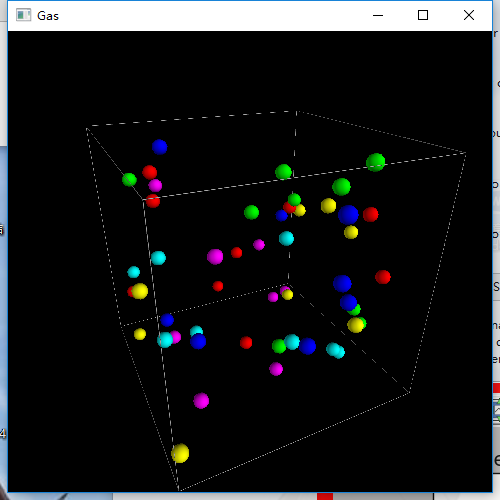
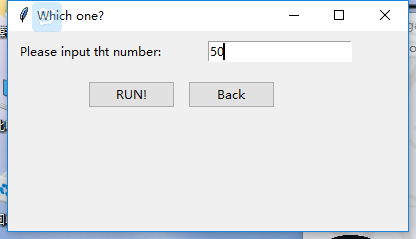
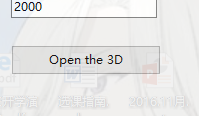
Specific data and save



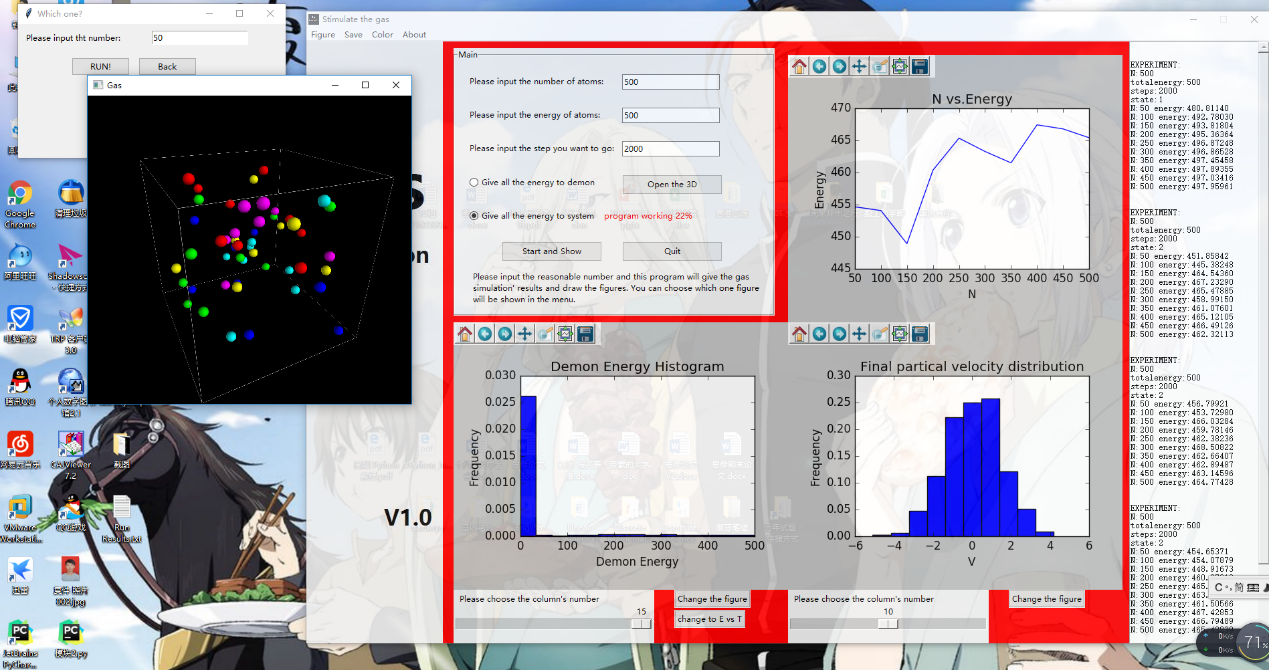
Save figure



About

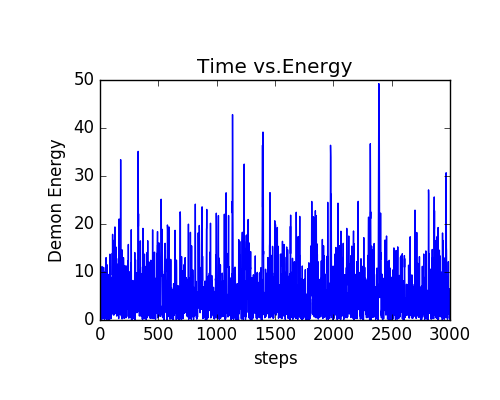
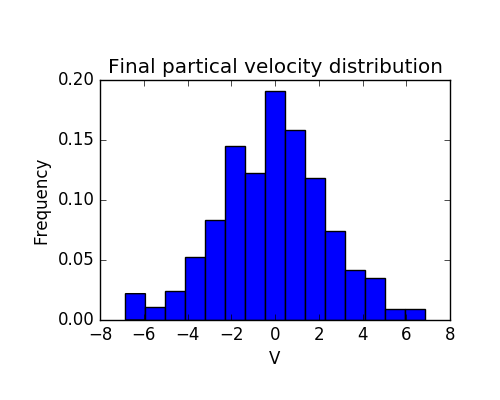
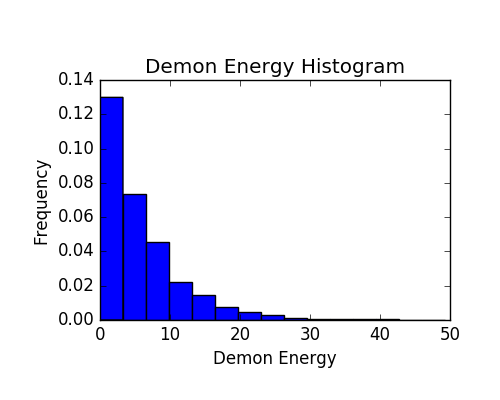
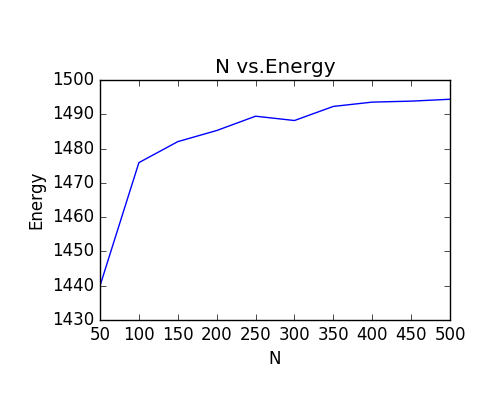


Show 3D animation

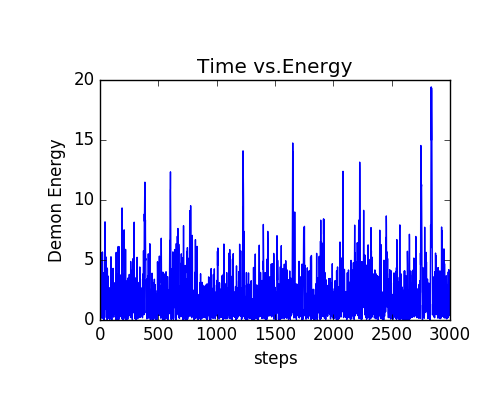
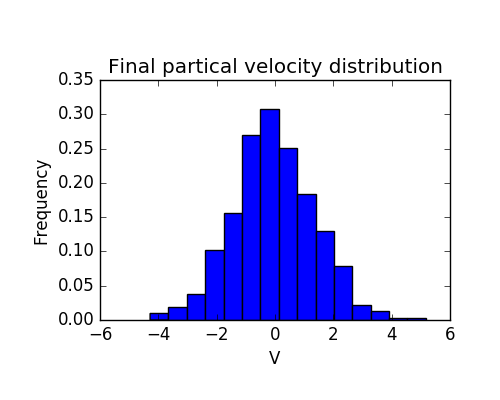
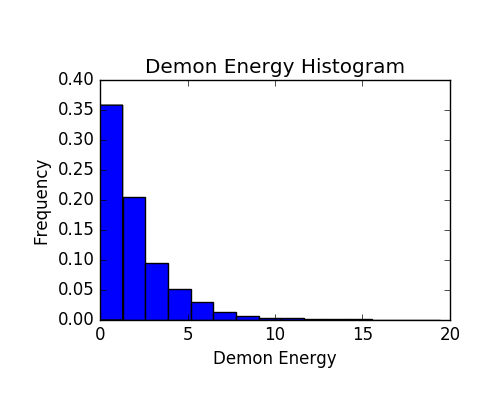
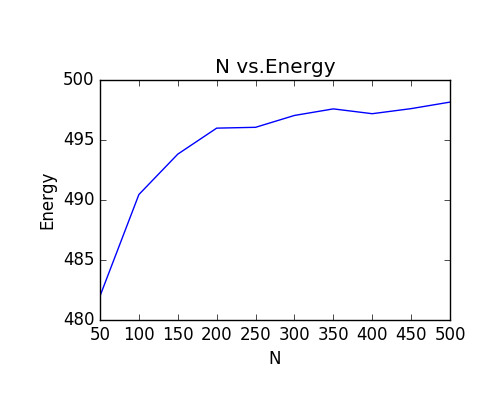


3D animation and 2D calculation are different threads

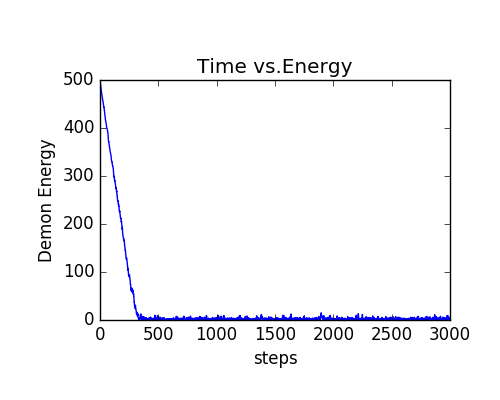
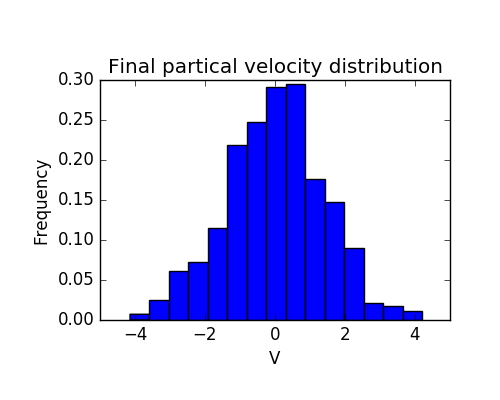
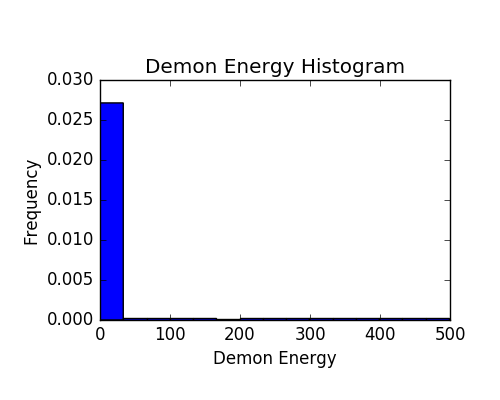
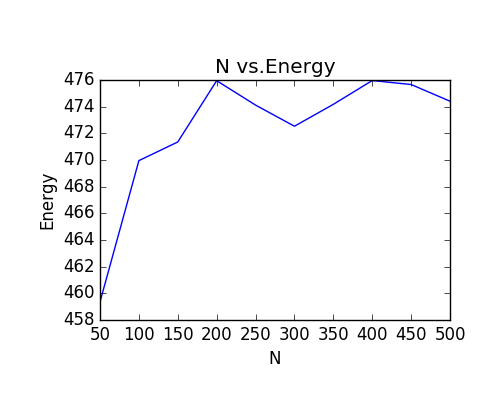
**5.2 Testing Procedure, Data and Result**

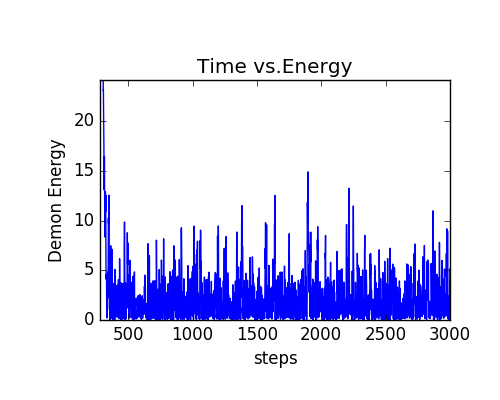


Result of the experiment N=500 total energy=1500 steps=3000 state=1

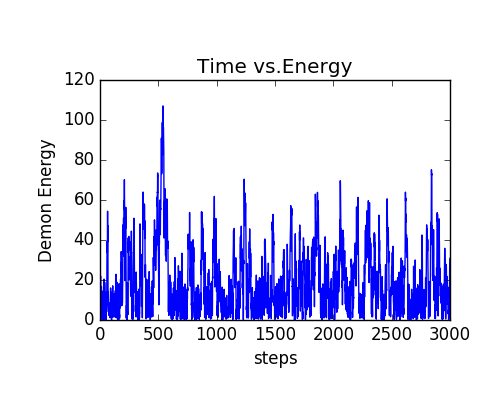
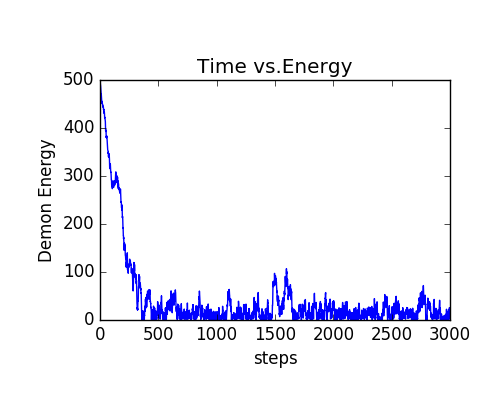


Result of the experiment N=500 total energy=500 steps=3000 state=1



(the graph after zoom)

Result of the experiment N=500 total energy=500 steps=3000 state=1

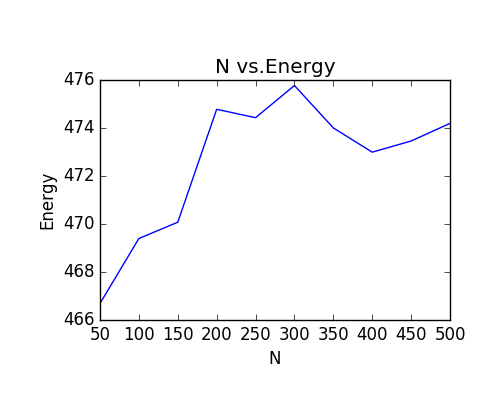
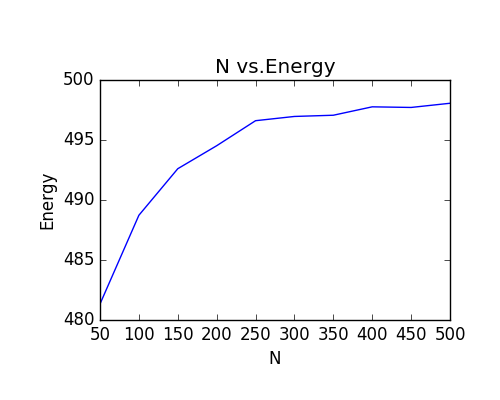
result when N=50 total energy=500 steps=3000 state=1 and 2

**6 Conclusion**

The discussion of your experiment and answers the questions listed above.

What is your research result? What are your experience and lesson on this project?

According to the results we get from the experiment. If all the energy is given to demon when initialize, the system energy is very close to the total energy, and it increases as the N increases. The relationship is between system energy and N is about **system energy = 0.0134ln(N) + 0.9681.** We think it has something to do with the entropy theory. The two plot is very different between two states. In the second state, the system energy is farther than that in state 1. And it don’t have a very smooth curve.



Graphs in state

From the histogram Final-Particle-Velocity. We think the distribution of the particles is like normal distribution. And the velocities in state1 which is equal to the initial one has a frequency in 15％ to 20％。

From the histogram Demon-Energy, We think the distribution is like exponential distribution, and comparing to state1, In state 2 the Demon energy is decreasing rapidly toward 0 so the span of the x axel is 0-500 and the histogram will seems more focus.

In the graph Demon-energy-time, the demon energy is decreasing and be more likely to be very low as N increases. As in the result when N is 500 demon energy is more likely to be below 15, comparing to 80 when N is 50. And in state 2, the speed of dropping is smaller when N is small. But no matter what the state is and how large the N is. The energy is always waving along the time.

From this project, we gained a more complete knowledge about python and more practical experience of programming. We learned small knowledge about threading, GUI building, and using of Object Oriented programming. Also we have experienced the procedure of a project, and an experiment. We got more knowledge about how computer simulate and about the demon algorithm. And we have learned that the drawbacks of using global variables

Besides the programming knowledge, we have experienced how to cooperate with others, what’s the best way to combine two man’s effort, which will be very useful in the future.