Project 1 – Visualize ODE With SciPy

Asher Shores

Grand Canyon University

CST-305 - Principles of Modeling and Simulation Lecture & Lab

Dr. Ricardo Citro

September 19, 2021

**Responsibilities and Completed Tasks**

This project sought to take a real world problem and simplify it into a working model which in turn could be represented by a mathematical model. This then could be translated into a computation model and use Python Libraries to simulate the real world problem accurately. In practice, this meant a Python script that could take a representative ODE and generate a visual representation of potential solutions. The ODE, the “real world problem” must be based on a performance metric of a computerized system. Thereby, taking an ODE as input and producing solution visuals as the output.

**System Performance Context Description**

When working within the context of communication networks, a simple example of a performance metric is throughput. Throughput can be defined as “the rate of production or the rate at which something is processed.” In general, this can be expressed as some unit, be it material or data, per unit of time.

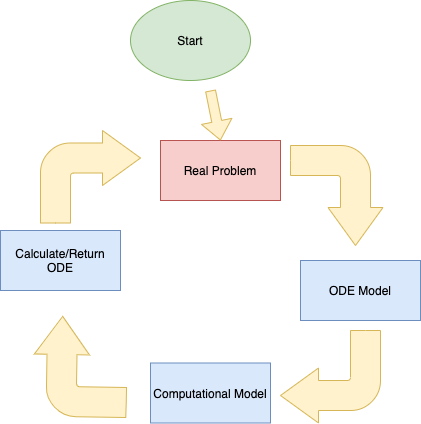
**Specific Problem Solved**

This metric being fairly simple, the problem to be solved is how much throughput, the rate of material being processed, is generated with varying amounts of materials over time.

**Mathematical Approach for Solving**

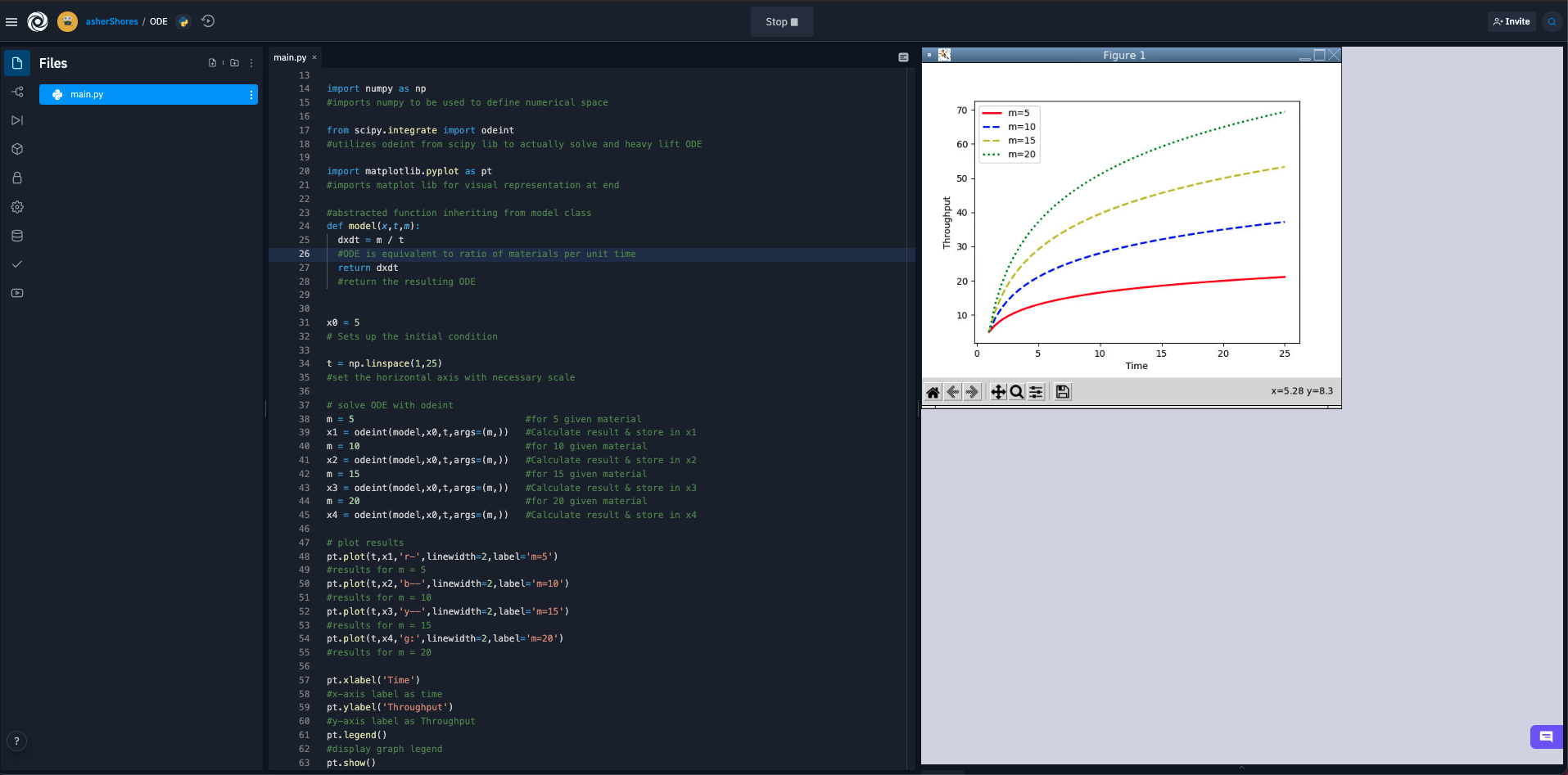
Assigning throughput as some tangible variable x, we can find x by dividing the material m by time. Thus, we have dx/dt = m/t. Assuming we will be dealing with a normal period of time, our rate will change by increasing material and consequently the throughput.

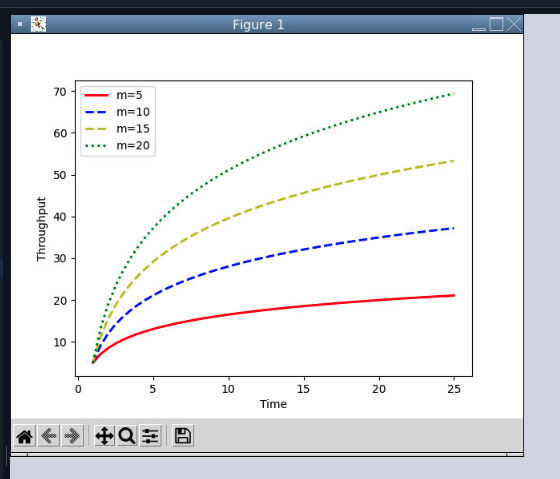
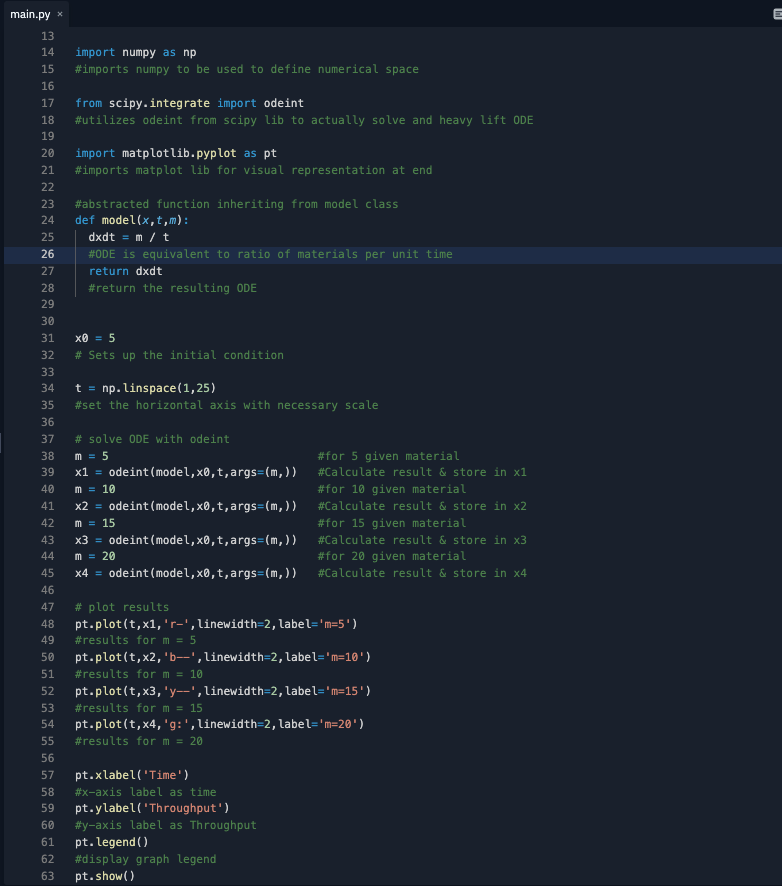
**Implementation in Code**

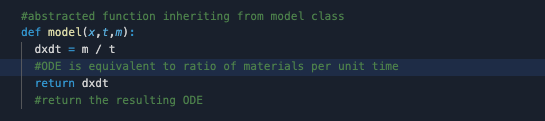


To implement the ODE into a python script, certain steps, especially preparatory ones were needed. An environment to code in was first necessary, such as pycharm, VS Code, or Replit, this makes the process much easier and helps protect against bugs along the way. Additionally, libs necessary for the program to function would need to be downloaded and installed properly. These libs include numpy, scipy, and matplotlib. To actually recreate the model in code, a mathematical model must be defined and then solved using ODEint and finally plotted using matplotlib.

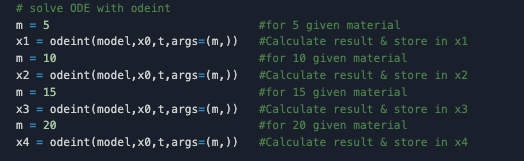
**Screenshots Depicting Code Running**

****

****

**Key Phrases**

Defining the model and returning it.



Solving the ODE with odeint and storing data for use in graphing.

**GitHub Documentation**

<https://github.com/asherShores5/AShores---Visualize-ODE-with-SciPy>

References

Burke, John. “What Is Throughput? - Definition from Whatis.com.” SearchNetworking, TechTarget, 5 May 2015, www.techtarget.com/searchnetworking/definition/throughput.

“Dynamics and Control.” Solve Differential Equations with ODEINT, apmonitor.com/pdc/index.php/Main/SolveDifferentialEquations.

“Scipy.integrate.odeint¶.” Scipy.integrate.odeint - SciPy v1.7.1 Manual, docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.odeint.html.