

# Solving Dynamical Systems in Python

Cody Riley

8/10/2022

# Chapter 1

## The Development Phase

### 1.1 Stages of Development

The development process of my dynamical system solver was relatively straightforward in the early stages of development. The first stage of development was to build a solid plan for what I planned to make. As discussed in my statement of intent, I wanted to develop an easy-to-use piece of software that would allow any user to input a set of first-order differential equations and have the python script approximate a solution to the corresponding dynamical system. In the early stages of planning, I thought it would be good to allow the user to input how many equations they would like to include in the solver. However, although the final build allows this to occur, it is more likely for bugs to occur, and the app becomes very tedious to use.

At first, I thought building such a program would be relatively straightforward, and I assumed I would have no issues. However, it proved more difficult than I anticipated, as I had to ensure that the correct number of equations were being generated alongside the number of variables. The name of the variables was also a challenge, and for simplicity, during the early stages of development, I decided to use the variable naming convention of  $x_n$ , where  $n \in \mathbb{Z}^+$ .

### 1.2 Positive Outcomes

There were many positive outcomes in the development of my Dynamical Systems app. Firstly, I successfully allowed users to enter a system of  $n$  equations in the terminal version of the application.

## 1.3 Challenging Parts of Development

Handling discontinuities of functions a user inputs was one of the biggest challenges in the development of the Python application. For example, say you had a simple one-dimensional problem, equivalent to

$$\frac{dy}{dt} = \frac{1}{t}$$

over a timespan of  $[-5, 5]$  (this is just the domain we will plot over). In this case, we would encounter an error because our solver cannot solve the differential equation for  $t > 0$  and hence we are unable to plot an accurate solution to that part of the solution.

## 1.4 Future Developments for the Project

I plan to work on this project further past the deadline. I just like to have projects to do in my spare time, and there are a few features I will be adding to my project. Firstly, I would like to add some functionality that I ran out of time to add. The plan was to give the user the option to add as many equations as they would like to solve the dynamical system. This would result in a much more complex system being solved. This functionality works in the terminal version of the program, however, I just ran out of time to give the GUI the ability to change how many equations the algorithm had to solve. I would also like to add the ability for the user to clear the current plot showing on the GUI, as this would result in better accessibility to the overall use of the tool. For example, at the moment, the current build of the Dynamical Systems Solver App only allows for you to plot a single graph on the canvas, and after plotting this graph you would have to restart the app to plot the next one. I have tried a few approaches to this problem already,

## Chapter 2

# How to Use the Application