

2D Graph Plotter and Visualizer

Akash Lalitkumar Makwana, Atik Zakirhusen Mujawar, Lalit Shailesh Jain, Bhargavi Dalal, Smita Bansod

IT Department, Shah and Anchor Kutchhi Engineering College, Mumbai, Maharashtra, India

ABSTRACT

Article Info

Volume 7, Issue 4

Page Number: 01-04

Publication Issue :

July-August-2021

Article History

Accepted : 25 June 2021

Published : 02 July 2021

In many fields of science and engineering, we often encounter the problem of solving N linear equations of type $x + y$ and trigonometric equations. All technological, biological and social networks can be represented as graphs. Therefore, graphs are used in the research of new algorithms and protocols based on simulation in various fields of science. We aim to create a python-based graph generator that will draw any type of equation i.e., linear, algebraic, trigonometric and logarithmic on the graph. More importantly, the application is designed to draw multiple graphs on the same canvas and then analyse the results. We have included one more module where user can upload a CSV file consisting of raw and get a desired pie chart, line graph as well as bar graph. Our system mainly focuses on generating an output for a later analysis by downloading the graphs they have plotted.

Keywords : Graphs, Python, Matplotlib, Numpy

I. INTRODUCTION

In math, a graph can be defined as a pictorial representation or a diagram that represents data or values in an organized manner. Graph creator is tool which help students to create a graph by just specifying the mathematical equation. Plotting and graphing are methods of visualizing the behaviour of mathematical functions. It helps student to analyse different and complex mathematical equations using pictorial diagram rather than solving it manually using much of the steps.

Python is a suitable language for both mastering and real global programming.[1] In our proposed system we have implemented this project using python and used its libraries. We have used matplotlib and

numpy as python libraries. Matplotlib is a low-level graph plotting library in python. It is a comprehensive library for developing static, lively, and interactive visualizations in python. In this project we have two main module i.e analytical and graphical modules. In analytical module there are three option such as bar graph, pie chart, line graph, user can create this by uploading a CSV file user will gets its desired output. In graphical module user can create various graph in one canvas different types of graph can be plotted using various equation such as trigonometric, logarithmic, algebraic, exponential, etc. Also, another feature is that user can plot multiple graphs on same canvas and download its output.

II. LITERATURE REVIEW

In this research [6] the main objective was to solve the linear equation problem of the form $A\theta = b$ by using distributed algorithms under a general directed graph. The research mainly consists of two algorithms i.e. "consensus + innovation" distributed algorithm for solving linear equation under a directed graph and derived "consensus + residual" distributed algorithm. In summary, the main contribution of this paper is to propose a dual "consensus + innovation" algorithm for solving the problem of linear equations under a directed graph, and a "consensus + residual" algorithm under a directed graph use the residual to represent the term, and compare the rates of the two algorithms through the simulation results.

This paper [4] which we referred was a complete system of creating, drawing graphs and using methods to portray it to the learners. Computer Algebra was the technology used, and advanced mathematical functions came into picture as technique to build the system. True color-screening, touch-pen operations was some of its brilliant features. Mentioning of co-ordinates for the graph to be drawn contributed to its drawback.

In this research [2] the main objective was Various graph-based information theoretic measures exist, including those accounting for the spectral distribution of graph matrix and those defining distributions on the graph vertices. A method for computing the structural complexity of graphs based on their signal transformation has been introduced. In this research [2], they proposed to compute graph information theoretic measures of functional connectivity networks based on the normalized power spectrum of graph signals, introducing a new definition of graph entropy and graph divergence. It uses technology as Multi-objective optimization, generic algorithm. And technique as Parallelization, multi-objective, scattered, star coordinate graph,

petal diagram. It features or advantages are Accuracy, easy comparing, performance indicator, auto-plotting. In this paper [3] one of the python modules – matplotlib, was used as technology. Here main focus was given to create highly customizable and reusable graphs. Some of the techniques which were used where Network Graph Topologies and Matlab. They added a great value in creating graphs, basically one graph at a time, which also can be pointed as a kind of drawback that they created only one graph at a time.

The main objective in this paper [5] was to solve the problem of solving a system of N linear equations $Ax = b$ (where A is a known $N \times N$ matrix, and b and x are, respectively, the known and unknown $N \times 1$ vectors). So, the paper presents a efficient parallel algorithm to solve the above problem on a star graph. The algorithm basically solves linear system of equations on the star graph also the above algorithm has two shortcomings it takes twice the amount of memory than that of the AD algorithm [5], and it supports only partial pivoting.

In existing system many of the available graph generator takes input i.e., equation from the user one at a time to create a graph.

Some of the graph generator tools require coordinates to be specified. The equation given by user is processed and after that visualization of the graph is deployed on the canvas i.e., one equation, one graph at same time.

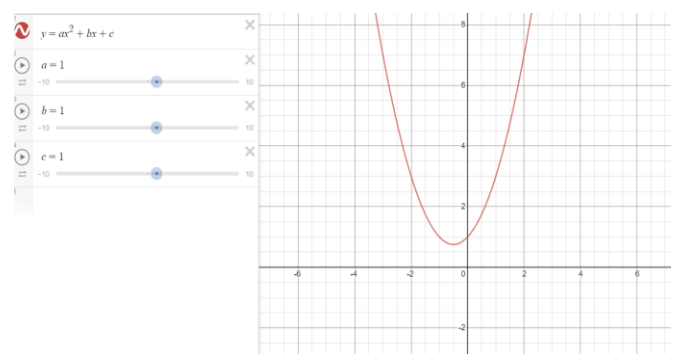


Fig. 1. Graph Analysis

III. PROPOSED SYSTEM

Our automation-based python project will be able to plot graphs for multiple equations at a time on same canvas.

The user does not need to solve the equation manually then find at least two coordinates, he just need to enter the equation and rest of the part from finding coordinates to deploying the graph will be automated.

Unlike other online graph makers, canvas isn't complicated or time-consuming. There's no learning curve – you'll get a perfect graph or diagram as and when you enter the equation, turning raw data into something that's both visual and easy to understand. One of the feature in our system also enables user to visualize the raw data needed for study and analysis using line, bar graph and pie chart as well, this can be done by uploading the CSV file consisting the data and getting the graphs and charts drawn from it.

IV. WORK FLOW

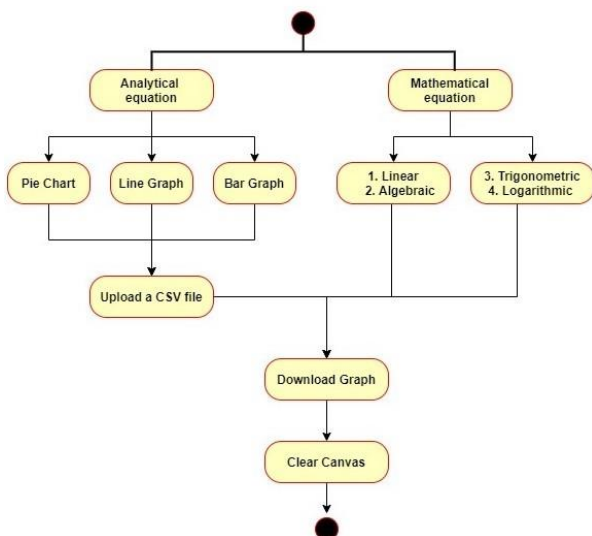


Fig. 2. System Flow

Modules

Input Module - In input module basically we provide user a input section, which will allow user to provide equation for graph which is to be generated.

Equation Translation Module - In this module the system will solve the user equation for one variable in terms of other (say y in terms of x) i.e., $y =$ (equation of variable x)

Equation processing module - In this module system will process the equation, find out the coordinates, make a list of all possible values between the two variables.

Display Module - After receiving the input from the above module, display module will deploy it on the visualization platform i.e., canvas available.

V. IMPLEMENTATION

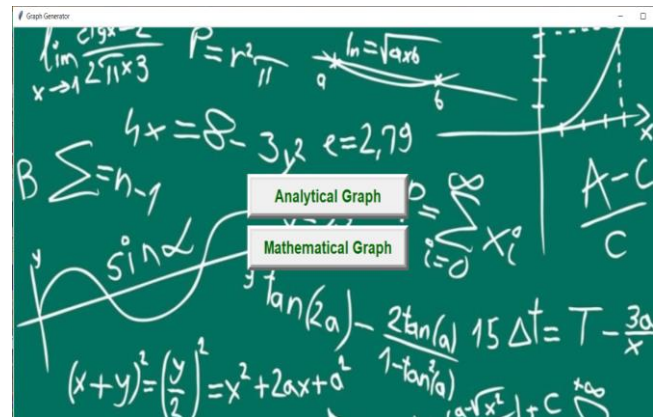


Fig. 3. Graph Generator

These are the two main modules

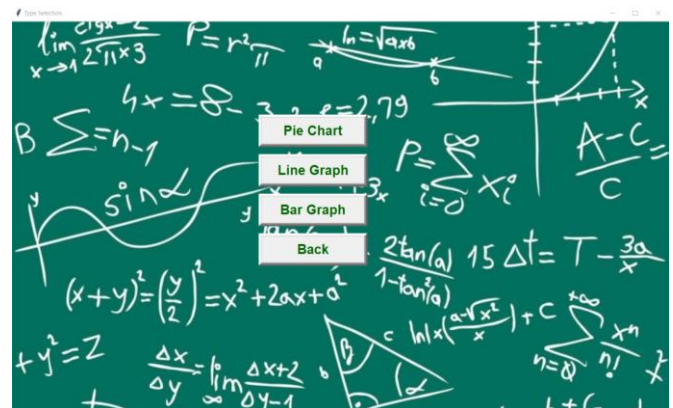


Fig. 4. Type Selection

Analytical Module includes Pie chart, Line, and Bar Graph. Here we can upload the CSV file which includes raw data, which is then represented using charts and graphs.

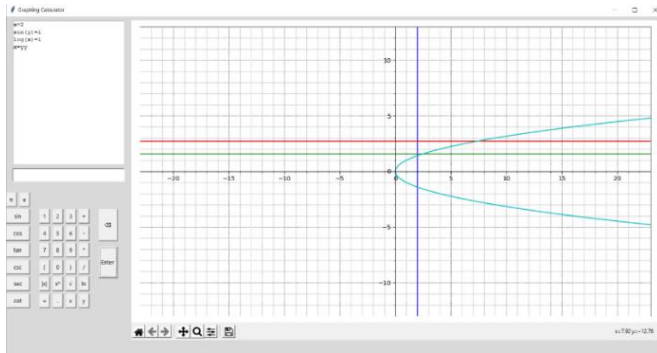


Fig. 5. Graph Calculator

This is the Mathematical module of our project, here we can plot various graphs of different types of equations i.e., Algebraic, Logarithmic, Exponential, etc. It includes a feature where we can plot any number of graphs on the same canvas, which can be seen in the snapshot above. Above canvas can also be downloaded.

VI. SUMMARY

This program is a visual workspace that combines diagramming, data visualization, and collaboration to accelerate understanding and drive innovation. One of its prominent perks is the real time modification in the curves of various graphs as you go on varying the input parameters. This technology is simple, fast and people will be satisfied after using this. It will be a window application which will be designed using different and much powerful libraries of python along with the automation.

VII. ACKNOWLEDGEMENT

The satisfaction that accompanies that the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. We are grateful to our project guide Ms.Smita Bansod and Ms Bhargavi Dalal for the guidance, inspiration and constructive suggestions that helpful us in the preparation of this project.

VIII. REFERENCES

- [1]. JULIAN BLANK AND KALYANMOY DEB “Pymoo: Multi-Objective Optimization in Python,” April 27, 2020.
- [2]. Vasileios Vlachos, Theodoros Siklaidis, Kyriaki Chantzi “GRATIS: A GRaph Tool for Information Systems Scientists” 2019 IEEE.
- [3]. Jun Nie1 Tianying Chen, Hongguang Fu “Dynamic Geometry in Graphing Calculator” 2009 IEE
- [4]. Ramesh Chandraa and C. Siva Ram Murthy “A faster algorithm for solving linear algebraic equations on the star graph “ 7 February 2003
- [5]. Yu Tang, Jie Mei, “Distributed Algorithms for Solving a Linear Equation Under a Directed Graph”, 2018

Cite this article as :

Akash Lalitkumar Makwana, Atik Zakirhusen Mujawar, Lalit Shailesh Jain, Bhargavi Dalal, Smita Bansod, "2D Graph Plotter and Visualizer", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 4, pp. 01-04, July-August 2021. Available at doi : <https://doi.org/10.32628/CSEIT2173173>
Journal URL : <https://ijsrcseit.com/CSEIT2173173>