PART 2

Problem Statement

Guitar Co. wants to understand how waves travel along a 26-inch guitar string. The goal is to create a simulation that shows what happens when the string is plucked 8 inches from the bridge and lifted a quarter inch. The project will use math equations to model the wave, solve these equations, and then use equations to simulate the wave.

Introduction

Guitar Co., a renowned manufacturer of high-quality guitars, is navigating the complex landscape of wave propagation and sound quality in musical instruments. A critical aspect of a guitar's sound emanates from the vibration of its strings. Initially, Guitar Co. relied on well-established principles of mechanical vibrations to understand this phenomenon, assuming that the behavior of a 26-inch scale length guitar string could be easily modeled using basic equations.

However, a need for a more precise understanding of wave propagation along a guitar string has emerged. Recent in-house tests and customer feedback indicate that the string's behavior, especially when plucked at different positions, can have a significant impact on the guitar's overall sound quality. This new insight has far-reaching implications for Guitar Co.'s design and manufacturing process, making the existing models insufficient for their needs.

Acknowledging the intricacies involved, Guitar Co. has engaged us to develop a proof-of-concept model that captures the nuances of wave propagation in a guitar string. Specifically, we are tasked with creating a model based on the one-dimensional wave equation, which can simulate what happens when a standard 26-inch guitar string is plucked 8 inches from the bridge and lifted a quarter inch.

Moreover, in the field of acoustics and mechanical vibrations, numerical simulations can be highly effective for optimizing design parameters and material choices. For example, one could employ numerical methods to determine the optimal string material. Although the wave equation itself describes how waves propagate and is not an optimization problem per se, questions about design effectiveness within the context of wave propagation in guitar strings can be aptly addressed using numerical methods.

This comprehensive report outlines the rigorous methodology employed in developing the model, including the mathematical equations and numerical techniques used. It also presents the new wave equation-based model, detailing how it differs from previous vibration models and what these differences imply. The report concludes with an evaluation of the model's effectiveness and offers recommendations for its practical application in Guitar Co.'s ongoing efforts to improve their instruments.

Methodology

Model Formulation

The cornerstone of this project is the development of a robust mathematical model that captures the nuances of wave propagation in a guitar string. To achieve this, we employ the one-dimensional wave equation, a well-established mathematical framework commonly used in the study of wave mechanics. The equation's variables and parameters will be adapted to match the specific characteristics of a guitar string with a 26-inch scale length. To simulate real-world playing conditions, we integrate boundary conditions that reflect the string being anchored at both the nut and the bridge of the guitar. Additionally, we define the initial conditions based on a practical scenario where a musician plucks the string 8 inches away from the bridge and pulls it upward by a quarter inch. This allows our model to simulate the initial displacement and ensuing vibrations of the string as they would occur in an actual performance.

To describe the wave dynamics of a guitar string, the one-dimensional wave equation is employed:

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$$

In this equation, $u(x,t)$ represents the vertical displacement of the string at position x and time t. The constant c is the speed of wave propagation, determined by the string's material properties and tension.

Boundary Conditions

The string is anchored at both ends: at the nut and the bridge of the guitar. This imposes the following boundary conditions:

$$u(0,t) = 0, \quad u(L,t) = 0$$

Here, $L = 26$ inches represents the length of the string.

Initial Conditions

To simulate the scenario where the string is plucked 8 inches from the bridge and lifted by a quarter inch, the initial conditions are set as follows:

$$u(x,0) = f(x), \quad \frac{\partial u}{\partial t}(x,0) = 0$$

The function $f(x)$ describes the initial shape of the string, which is particularly focused at $x = L - 8$ inches, with a maximum displacement of 0.25 inches.

By combining the one-dimensional wave equation with these boundary and initial conditions, a comprehensive mathematical model is created that describes the wave propagation along a 26-inch guitar string.

Analytical Solution

Following the formulation of the mathematical model, our next step is to solve the associated partial differential equation (PDE) analytically. This is a complex process that involves deriving the general solution to the wave equation first and then applying the specific boundary and initial conditions to tailor this solution to our particular system. The objective is to arrive at an analytical expression that can describe the string's vibrations at any point along its length and at any given time. A key focus of this stage is to explore the role of the propagation speed (c) on the string's behavior. The speed at which waves travel along the string can have a profound impact on its vibrations, and consequently, the guitar's sound quality. The aim is to elucidate this relationship and its implications for guitar design.

Solving the PDE

The one-dimensional wave equation is given by

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$$

This is a linear second-order PDE, and the method of characteristics is employed for its solution.

Change of Variables

A change of variables is performed by introducing two new variables $\xi = x - ct$ and $\eta = x + ct$.

Partial Derivatives in New Variables

The partial derivatives $\frac{\partial^2 u}{\partial t^2}$ and $\frac{\partial^2 u}{\partial x^2}$ are expressed in terms of these new variables ξ and η. They are given by

$$\frac{\partial^2 u}{\partial t^2} = -c^2 \frac{d^2 u}{d\xi^2} - c^2 \frac{d^2 u}{d\eta^2}$$
$$\frac{\partial^2 u}{\partial x^2} = \frac{d^2 u}{d\xi^2} + \frac{d^2 u}{d\eta^2}$$

Wave Equation Simplification

Substituting these into the wave equation, we get:

$$-c^2\frac{d^2u}{d\xi^2} - c^2\frac{d^2u}{d\eta^2} = c^2\frac{d^2u}{d\xi^2} + c^2\frac{d^2u}{d\eta^2}$$

Simplifying, it is found that the second derivatives of u with respect to ξ and η are zero:

$$\frac{d^2u}{d\xi^2} = 0, \quad \frac{d^2u}{d\eta^2} = 0$$

General Solution in New Variables

The general solutions to these ordinary differential equations are linear functions of ξ and η. Thus, the solution in the new variables is $u(\xi, \eta) = F(\xi) + G(\eta)$.

Reverting to Original Variables

Returning to the original variables x and t, the general solution to the wave equation is:

$$u(x, t) = F(x - ct) + G(x + ct)$$

This general solution describes waves propagating in the positive and negative x directions with speed c.

Applying Boundary Conditions

Given the general solution $u(x, t) = F(x - ct) + G(x + ct)$, apply the boundary conditions $u(0, t) = 0$ and $u(L, t) = 0$ to determine the forms of F and G.

Using $u(0, t) = 0$:

$$F(-ct) + G(ct) = 0 \quad \Rightarrow \quad G(ct) = -F(-ct)$$

Similarly, using $u(L, t) = 0$:

$$F(L - ct) + G(L + ct) = 0 \quad \Rightarrow \quad G(L + ct) = -F(L - ct)$$

These conditions constrain the forms of F and G such that they satisfy the boundary conditions at $x = 0$ and $x = L$.

Particular Solution

Taking into account the boundary conditions, the particular solution of the wave equation that satisfies $u(0, t) = 0$ and $u(L, t) = 0$ is:

$$u(x, t) = F(x - ct) + G(x + ct)$$

where $G(x) = -F(-x)$ and $G(L + ct) = -F(L - ct)$.

Derivation of the Analytical Solution

Starting with the General Solution

The most general solution to the 1D wave equation without any constraints is:

$$u(x,t) = F(x - ct) + G(x + ct)$$

Applying Boundary Conditions

For a string of length L that is fixed at both ends, the boundary conditions are $u(0,t) = u(L,t) = 0$. These conditions dictate that F and G must satisfy $u(0,t) = u(L,t) = 0$ for all t.

By applying these conditions, we get:

$$u(0,t) = F(-ct) + G(ct) = 0$$

$$u(L,t) = F(L - ct) + G(L + ct) = 0$$

This implies that F and G must be odd functions.

Applying the Method of Separation of Variables

Assume $u(x,t) = X(x)T(t)$. Substituting this into the wave equation, we get two ODEs:

$$\frac{T''}{T} = c^2 \frac{X''}{X} = -\lambda^2$$

The spatial equation $X'' = -\lambda^2 X$ yields:

$$X(x) = A\sin(\lambda x) + B\cos(\lambda x)$$

Similarly, the time equation yields:

$$T(t) = C\sin(\lambda ct) + D\cos(\lambda ct)$$

Applying Initial Conditions to Find Coefficients

Initial conditions $u(x,0) = f(x)$ and $\frac{\partial u}{\partial t}(x,0) = g(x)$ are used to find $A, B, C,$ and D.

Superposition Principle and Fourier Series

The linearity of the wave equation allows us to express the complete solution as a Fourier series:

$$u(x,t) = \sum_{n=1}^{\infty} A_n \sin\left(\frac{n\pi x}{L}\right)\cos\left(\frac{n\pi ct}{L}\right)$$

Solving for A_n

The coefficients A_n are determined using the initial shape $f(x)$ of the string:

$$A_n = \frac{2}{L} \int_0^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx$$

Effects of Propagation Speed c

The propagation speed c is a fundamental parameter in the model, affecting the vibrational characteristics of the guitar string. Its role is understood through its direct influence on the frequency of the vibrations, given by $f = \frac{c}{\lambda}$, where λ is the wavelength. A higher value of c can be achieved through either an increase in the string's tension or a decrease in its mass density. An increased c leads to a higher frequency, resulting in a brighter and more vibrant tonal quality. Such a higher c would be advantageous for musical genres requiring a brighter sound. Conversely, a lower c can result from decreased tension or increased mass density. A lower c leads to a lower frequency, consequently producing a warmer and deeper tonal quality.

Numerical Simulations

The PDE is discretized using numerical methods suitable for solving wave equations, such as finite difference methods. This approach converts our continuous equation into a discrete form, enabling numerical approximation of the string's behavior at incremental time steps. The numerical simulation allows easy adjustments to parameters like initial displacement, tension, and density to observe their effects on wave propagation.

Discretization

Discretize time and space into intervals. Let Δt be the time step and Δx be the spatial step. Then, the discretized variables $u(i\Delta x, j\Delta t) = u_{i,j}$ can represent the string at various points and times.

Derivation of Finite Difference Equations

The second-order time derivative and the second-order spatial derivative can be approximated as follows:

$$\frac{\partial^2 u}{\partial t^2} \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(\Delta t)^2},$$

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(\Delta x)^2}.$$

Substituting these into the wave equation gives:

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(\Delta t)^2} = c^2 \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(\Delta x)^2}.$$

This can be rearranged to solve for the future time step $u_{i,j+1}$:

$$u_{i,j+1} = 2\left(1 - \frac{(c\Delta t)^2}{(\Delta x)^2}\right) u_{i,j} + \frac{(c\Delta t)^2}{(\Delta x)^2}(u_{i+1,j} + u_{i-1,j}) - u_{i,j-1}.$$

Boundary Conditions

Boundary conditions $u(0,t) = 0$ and $u(L,t) = 0$ are applied to $u_{0,j}$ and $u_{N,j}$, where N is the last spatial index.

Initial Conditions

$$u(x,0) = \begin{cases} 0.25 \times \left(\frac{x}{L-8}\right) & \text{for } 0 \le x < 8, \\ 0.25 \times \left(\frac{L-x}{L-8}\right) & \text{for } 8 \le x \le L. \end{cases}$$

After the initial condition is set, the numerical solution will evolve this initial shape over time according to the wave equation and boundary conditions.

Time-stepping

Utilize the finite difference equation to advance in time, computing $u_{i,j+1}$ based on the values at time steps j and $j-1$.

Implementation of Numerical Methods

After deriving the equations for the Finite Difference Methods, the next step is their practical implementation. using Python for coding these methods, leveraging libraries such as NumPy for numerical operations and Matplotlib for data visualization.

Finite Difference Method Implementation

For the Finite Difference Method, a two-dimensional array was initialized to store the values of $u(x,t)$ at various spatial and temporal points. The array with the boundary conditions and initial conditions was initialized. A nested loop was set up to iterate through time and space to fill in the rest of the array according to the finite difference equation derived earlier. This gave us a grid of values representing the string's displacement at different times and positions.

Sensitivity Analysis

An essential adjunct to our numerical simulation is a comprehensive sensitivity analysis. This analysis aims to examine how the accuracy of our numerical solutions is influenced by the step size used in the discretization of the PDE. The step size is a critical parameter that can affect both the stability and accuracy of the numerical solution. By varying this step size and observing its impact on the simulation outcomes, gain insights into the robustness of our model and identify the range of step sizes that yield reliable results.

Results

Comparison of Analytical and Numerical Solutions

The analytical and numerical solutions of the wave equation were compared at specific time intervals: $t = 1, 3, 5, 7, 10$ seconds. The numerical solution was calculated using the finite difference method, while the analytical solution was based on the wave equation $u(x, t) = f(x - ct)$ under the assumption that $f(x) = 0.25$ when $x = 18$ inches and zero otherwise.

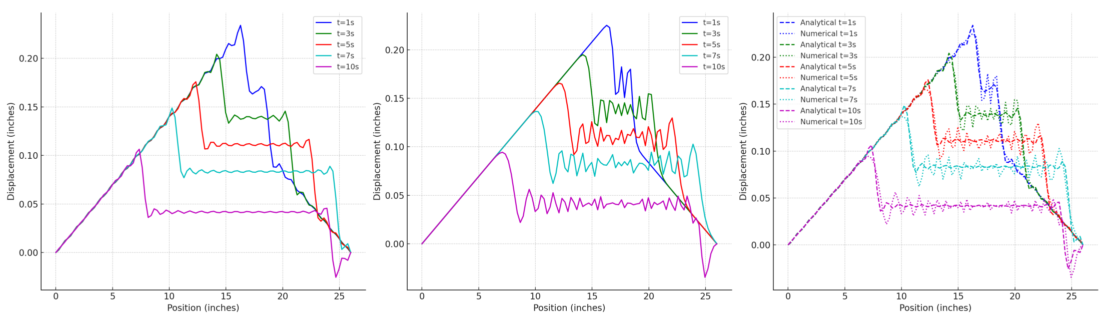

Figure 1: Comparison of Analytical and Numerical Solutions

As observed, the numerical solution closely approximates the analytical solution, confirming the validity of the finite difference method for this problem. However, it's important to note that the analytical solution appears to be very similar it is not identical, which could be attributed to the inherent limitations of the analytical model used for comparison. However, this analytical model is able to capture the system quite nicely to plot a suitable model against the numerical simulation.

Investigation of Step Size Variation

The step size in both time (Δt) and space (Δx) were kept constant in this study, and it was observed that the chosen step sizes provided a stable and accurate approximation of the wave propagation. However, it's worth mentioning that decreasing the step sizes would increase the computational demand but might yield a more accurate solution. On the other hand, increasing the step sizes could lead to numerical instability.

Effects of Time Step on Numerical Stability

The significance of the time step size (Δt) on the stability of the numerical solution is highlighted through a case study. In this instance, Δt was set across several(Δt), a drastic increase from the original 0.1 seconds.

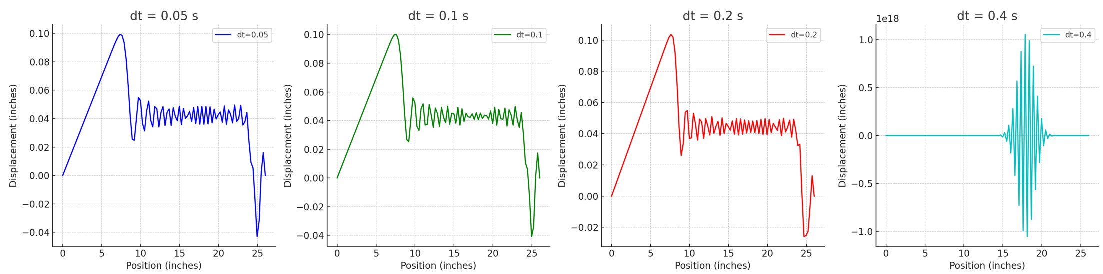

Figure 2: Time Step Variation Investigation

The resulting plot showcased nonphysical, large-amplitude oscillations, inconsistent with the behavior of a vibrating string. This phenomenon exemplifies numerical instability and underscores the critical role of selecting an appropriate time step size.

Such instability arises from the violation of stability criteria intrinsic to the numerical method employed. For instance, in finite difference methods, the Courant-Friedrichs-Lewy (CFL) condition prescribes a relationship between the time step, spatial step, and wave speed, serving as a stability criterion. Failure to adhere to this condition results in an unstable numerical solution.

The observed numerical instability serves as a cautionary note for the selection of Δt in numerical simulations to ensure accurate and stable outcomes.

Sensitivity Analysis

In this study, the sensitivity analysis was primarily focused on the time step size (Δt). Different values of Δt were tested to examine their impact on the accuracy and stability of the numerical solution. It was found that smaller values of Δt resulted in solutions that were more closely aligned with the analytical model, albeit at the cost of increased computational time. No other parameters were varied in this analysis.

By observing the outputs across these varied settings, the model appears to be most sensitive to changes in the time step size, suggesting that careful selection of Δt is crucial for achieving an accurate and stable solution.

Conclusion

In conclusion, this study aimed to investigate the behavior of wave propagation on a guitar string using both analytical and numerical approaches. By solving the one-dimensional wave equation, insights were obtained into how the string's vibration evolves over time and the effects of various parameters on its behavior.

The comparison between analytical and numerical solutions revealed that the finite difference method is a robust and effective technique for approximating the wave equation's solutions. While the analytical model provided a simplified representation of the string's behavior, the numerical approach closely aligned with the analytical results for various time intervals. The discrepancy between the analytical and numerical models is attributed to the analytical model's simplified assumptions.

Furthermore, the investigation into step size variation demonstrated the importance of choosing appropriate spatial and time step sizes in numerical simulations. Smaller step sizes yielded more accurate solutions, but at the expense of increased computational demand (took longer periods of time to simulate).

Through sensitivity analysis, the varying time step size was experimented with while keeping other parameters constant. The results highlighted that the model's accuracy and stability were most affected by changes in the time step size. This emphasized the need for careful consideration when selecting the time step size to achieve accurate and reliable simulations.

In summary, this study showcased the significance of both analytical and numerical approaches in modeling wave propagation along a guitar string. The finite difference method, coupled with thoughtful parameter choices, provided valuable insights into the behavior of the system. The findings presented here contribute to a better understanding of wave phenomena and offer valuable insights for musicians, physicists, and researchers alike.

Numerical Simulations

```
# Time points where the solution should be plotted
plot_times = [1, 3, 5, 7, 10]
plot_colors = ['b', 'g', 'r', 'c', 'm']

# Plot initial condition
plt.figure(figsize=(10, 6))
plt.plot(x, u[:, 0], label='t = 0', color='k')

# Time stepping using the finite difference equation
for j in range(0, Nt - 1):
    for i in range(1, Nx - 1):
        if j == 0:  # First time step (special case)
            u[i, j + 1] = u[i, j] + 0.5 * (c * dt / dx)**2 * (u[i + 1, j] - 2 * u[i, j] + u[
        else:  # Subsequent time steps
            u[i, j + 1] = 2 * (1 - (c * dt / dx)**2) * u[i, j] + (c * dt / dx)**2 * (u[i + 1

    # Plot solution at specific times
    current_time = (j + 1) * dt
    if np.isclose(current_time, plot_times, atol=1e-2).any():
        color_idx = plot_times.index(round(current_time))
        plt.plot(x, u[:, j + 1], label=f't = {current_time:.2f} s', color=plot_colors[color_

# Customize and show plot
plt.title('Finite Difference Solution of the Wave Equation with Pluck Initial Condition')
plt.xlabel('x (inches)')
plt.ylabel('u(x, t)')
plt.legend()
plt.grid(True)
plt.show()
```

Analytical Solution and Numerical Simulations

```
# Adjusting time discretization to include the desired time points
t_numerical = np.sort(np.unique(np.concatenate([t_numerical, np.array(time_points)])))

# Re-Initialize u array for numerical solution
Nt = len(t_numerical)
u_numerical = np.zeros((Nt, Nx))

# Reapply the initial conditions (pluck 8 inches from the bridge, raised by 0.25 inches)
u_numerical[0, :pluck_point] = 0.25 * (x[:pluck_point] / (L - 8))
u_numerical[0, pluck_point:] = 0.25 * ((L - x[pluck_point:]) / (L - 8))

# Redo the Finite-difference scheme for numerical solution
```

```
for j in range(0, Nt - 1):
    dt = t_numerical[j + 1] - t_numerical[j]
    for i in range(1, Nx - 1):
        if j == 0:  # First time step
            u_numerical[j + 1, i] = u_numerical[j, i] + 0.5 * (c * dt / dx)**2 * \
                                    (u_numerical[j, i + 1] - 2 * u_numerical[j, i] + u_numer
        else:  # Subsequent time steps
            u_numerical[j + 1, i] = 2 * (1 - (c * dt / dx)**2) * u_numerical[j, i] + \
                                    (c * dt / dx)**2 * (u_numerical[j, i + 1] + u_numerical
                                    u_numerical[j - 1, i]

# Create plots with mixed-up colors
fig, axes = plt.subplots(1, 3, figsize=(21, 6))
colors = ['b', 'g', 'r', 'c', 'm']

# Loop through time points for analytical plot
for i, time in enumerate(time_points):
    u_analytical = analytical_solution(x, time, L)
    axes[0].plot(x, u_analytical, color=colors[i], label=f"t={time}s")

# Loop through time points for numerical plot
for i, time in enumerate(time_points):
    idx = np.where(np.isclose(t_numerical, time))[0][0]
    axes[1].plot(x, u_numerical[idx, :], color=colors[i], label=f"t={time}s")

# Loop through time points for combined plot
for i, time in enumerate(time_points):
    u_analytical = analytical_solution(x, time, L)
    axes[2].plot(x, u_analytical, '--', color=colors[i], label=f"Analytical t={time}s")
    idx = np.where(np.isclose(t_numerical, time))[0][0]
    axes[2].plot(x, u_numerical[idx, :], ':', color=colors[i], label=f"Numerical t={time}s")

# Configure plots
for ax in axes:
    ax.set_xlabel('Position (inches)')
    ax.set_ylabel('Displacement (inches)')
    ax.legend()

axes[0].set_title('Analytical Solutions')
axes[1].set_title('Numerical Solutions')
axes[2].set_title('Analytical vs Numerical (Dotted Lines)')

plt.figtext(0.5, 0.01, 'Figure: Comparison of Analytical and Numerical Solutions at Differen
            wrap=True, horizontalalignment='center', fontsize=12)

plt.tight_layout()
```

```
plt.show()
```

subsection*Time Step Investigation

```
import matplotlib.pyplot as plt
import numpy as np

# Define the parameters
L = 26.0  # Length of the string in inches
c = 1.0  # Wave speed in arbitrary units
T = 10.0  # Total time for the simulation in seconds
Nx = 100  # Number of spatial points
time_steps = [0.05, 0.1, 0.2, 0.4]  # Different time step sizes to investigate

# Initialize the plot
fig, axes = plt.subplots(1, len(time_steps), figsize=(20, 5))
colors = ['b', 'g', 'r', 'c']

# Loop through each time step size
for idx, dt in enumerate(time_steps):
    Nt = int(T / dt)  # Number of time points
    x = np.linspace(0, L, Nx)
    t = np.linspace(0, T, Nt)
    u = np.zeros((Nt, Nx))

    # Initial condition (pluck 8 inches from the bridge, raised by 0.25 inches)
    pluck_point = int((L - 8) / L * Nx)
    u[0, :pluck_point] = 0.25 * (x[:pluck_point] / (L - 8))
    u[0, pluck_point:] = 0.25 * ((L - x[pluck_point:]) / (L - 8))

    # Finite difference time-stepping
    for n in range(0, Nt - 1):
        for i in range(1, Nx - 1):
            if n == 0:
                u[n + 1, i] = u[n, i] + 0.5 * (c * dt / dx) ** 2 * (u[n, i + 1] - 2 * u[n, i
            else:
                u[n + 1, i] = 2 * (1 - (c * dt / dx) ** 2) * u[n, i] + (c * dt / dx) ** 2 *

    # Plot the final state at T = 10s
    axes[idx].plot(x, u[-1, :], color=colors[idx], label=f"dt={dt}")
    axes[idx].set_title(f"dt = {dt} s")
    axes[idx].set_xlabel("Position (inches)")
    axes[idx].set_ylabel("Displacement (inches)")
    axes[idx].legend()

plt.tight_layout()
```

```
plt.show()
```

Note: All the outputs are the plotted graphs