Project Report
The Tacoma Narrows Bridge

Team Members:

- **Ankit Kumar Aggarwal**
- **Avinash Rajaraman Swaminathan**
- **Sayed Raheel Hussain**

Contents

Tacoma Narrow Bridge Accident

The 1940 Tacoma Narrows Bridge, the first Tacoma Narrows Bridge, was a suspension bridge in the U.S. state of Washington that spanned the Tacoma Narrows strait of Puget Sound between Tacoma and the Kitsap Peninsula. It opened to traffic on July 1, 1940, and dramatically collapsed into Puget Sound on November 7 of the same year. The Tacoma Narrows Bridge was built to replace an aging ferry system that connected the city of Tacoma with the Kitsap Peninsula. The bridge was designed by renowned engineer Leon Moisseiff, who also designed the Manhattan Bridge in New York City. The Tacoma Narrows Bridge was an innovative design for its time, with a relatively narrow deck and a suspension system that utilized thinner cables than previous suspension bridges.

However, just a few months after it opened, the bridge's deck began to oscillate and twist in the wind, leading to its eventual collapse. The disaster was a major setback for the engineering community and raised questions about the safety of suspension bridges. In the years that followed, new techniques were developed to better understand and account for the effects of wind on structures.

The collapse of the bridge, which was captured on film, was a major engineering disaster and led to new design and construction methods for suspension bridges. The cause of the collapse was attributed to a phenomenon known as aeroelastic flutter, which occurs when wind causes a structure to vibrate and oscillate at its natural frequency. The bridge's design and construction, as well as the failure to account for the effects of wind, were major factors in the disaster.

The Tacoma Narrows Bridge collapse was a major event in the history of engineering and helped to advance understanding of the effects of wind on structures.

Tacoma Bridge Collapses

TACOMA, Wash. . . . Like a strip of cloth rippling in the breeze, the center span of the Tacoma Narrows $6,500,000 bridge writhed and twisted under the force of a forty mile wind and collapsed into the channel below. This remarkable photograph was taken just as the bridge collapsed. Built with P.W.A. funds, an investigation into the causes of the collapse is now being made. Circle shows an automobile on the bridge. The driver crawled back to safety.

The Tacoma Narrows Bridge - Project Problem Statement

A mathematical model that attempts to capture the Tacoma Narrows Bridge incident was proposed by McKenna and Tuama [2001]. The goal is to explain how torsional, or twisting, oscillations can be magnified by forcing that is strictly vertical. Consider a roadway of width 2l hanging between two suspended cables, as in Figure 6.18(a). We will consider a two-dimensional slice of the bridge, ignoring the dimension of the bridge's length for this model, since we are only interested in the side-to-side motion. At rest, the roadway hangs at a certain equilibrium height due to gravity; let y denote the current distance the center of the roadway hangs below this equilibrium.

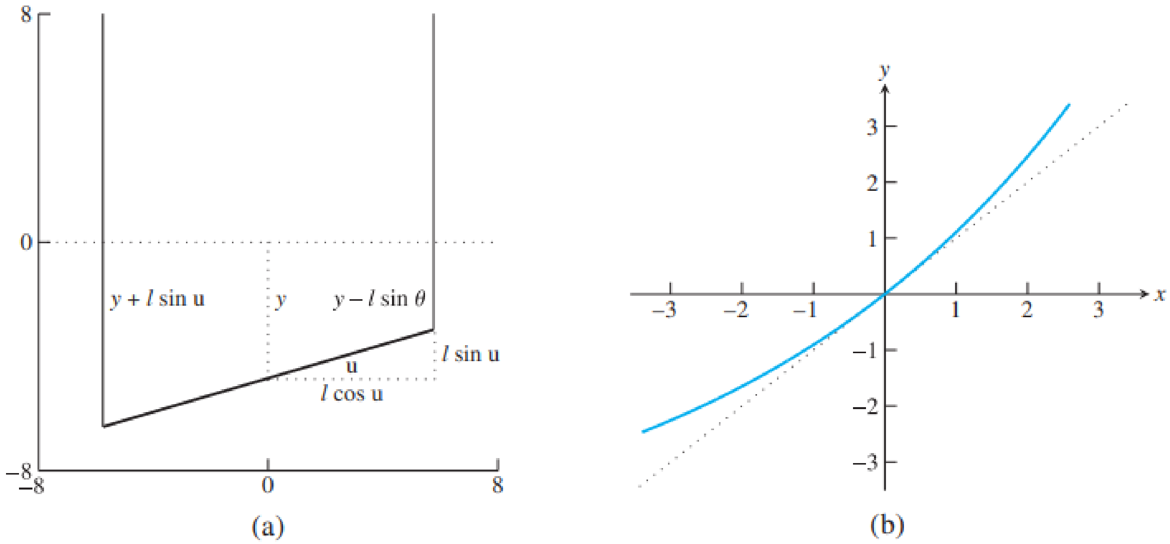


Figure 6.18 Schematics for the McKenna–Tuama model of the Tacoma Narrows Bridge. (a) Denote the distance from the roadway center of mass to its equilibrium position by y, and the angle of the roadway with the horizontal by θ. (b) Exponential Hooke's law curve $f(y) = (K/a)(e^{ay} - 1)$.

Hooke's law postulates a linear response, meaning that the restoring force the cables apply will be proportional to the deviation. Let θ be the angle the roadway makes with the horizontal. There are two suspension cables, stretched y − lsinθ and y + lsinθ from equilibrium, respectively. Assume that a viscous damping term is given that is proportional to the velocity. Using Newton's law F = ma and denoting Hooke's constant by K, the equations of motion for y and θ are as follows:

$$
\begin{aligned}
y'' &= -dy' - \left[\frac{K}{m}(y - l\sin\theta) + \frac{K}{m}(y + l\sin\theta)\right] \\
\theta'' &= -d\theta' + \frac{3\cos\theta}{l}\left[\frac{K}{m}(y - l\sin\theta) - \frac{K}{m}(y + l\sin\theta)\right].
\end{aligned}
$$

However, Hooke's law is designed for springs, where the restoring force is more or less equal whether the springs are compressed or stretched. McKenna and Tuama hypothesize that cables pull back with more force when stretched than they push back when compressed. (Think of a string as an extreme example.) They replace the linear Hooke's law restoring force f (y) = K y with a nonlinear force f (y) = (K/a)(eay − 1), as shown in Figure 6.18(b). Both functions have the same slope K at y = 0; but for the nonlinear force, a positive y (stretched cable) causes a stronger restoring force than the corresponding negative y (slackened cable). Making this replacement in the preceding equations yields

$$y'' = -dy' - \frac{K}{ma}\left[e^{a(y-l\sin\theta)} - 1 + e^{a(y+l\sin\theta)} - 1\right]$$
$$\theta'' = -d\theta' + \frac{3\cos\theta}{l}\frac{K}{ma}\left[e^{a(y-l\sin\theta)} - e^{a(y+l\sin\theta)}\right].$$

As the equations stand, the state y = y′ = θ = θ′ = 0 is an equilibrium. Now turn on the wind. Add the forcing term 0.2W sinωt to the right-hand side of the y equation, where W is the wind speed in km/hr. This adds a strictly vertical oscillation to the bridge.

1. Useful estimates for the physical constants can be made. The mass of a one-foot length of roadway was about 2500 kg, and the spring constant K has been estimated at 1000 Newtons. The roadway was about 12 meters wide. For this simulation, the damping coefficient was set at d = 0.01, and the Hooke's nonlinearity coefficient a = 0.2. An observer counted 38 vertical oscillations of the bridge in one minute shortly before the collapse—set ω = 2π(38/60). These coefficients are only guesses, but they suffice to show ranges of motion that tend to match photographic evidence of the bridge's final oscillations.

Tools and Technology

- Python
- Jupiter Notebook

Convergence and Function Testing

Convergence Rate Test

Empirically check that your Trapezoid and Runge-Kutta 4 solver achieves their theoretical convergence rate.

Solution: To empirically check that our numerical solvers achieve their theoretical convergence rate, we executed different solvers on a test problem with a known exact solution and then calculate the error between the numerical and exact solutions at different step sizes. The error is expected to decrease with decreasing step size, and the rate of decrease can be used to estimate the order of convergence of the solver.

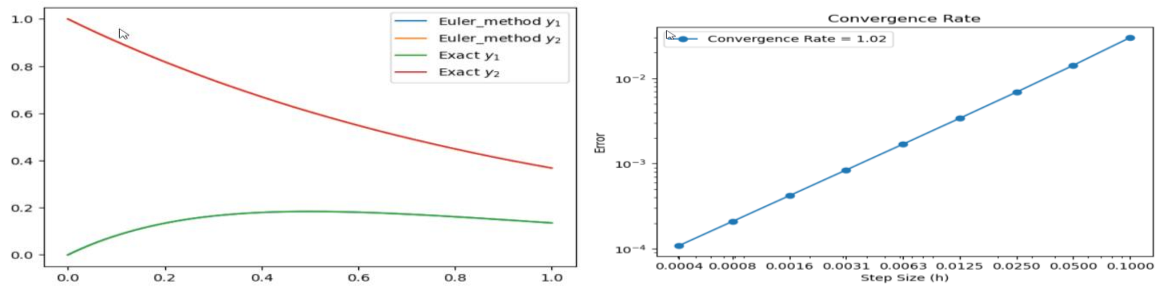
Approach:

- First, we defined, Simpler differential equation to check our solver.
- Then we defined functions for different numerical methods i.e., Euler's Method, Trapezoid Method, and Runga Kutta Order4 Method.
- Then we defined "check_convergence" function which will take different arguments and execute respective function to check and plot the convergence rate along with the step size and error information.

Here is the outcome of different solvers along with their convergence rate.

Euler's Method

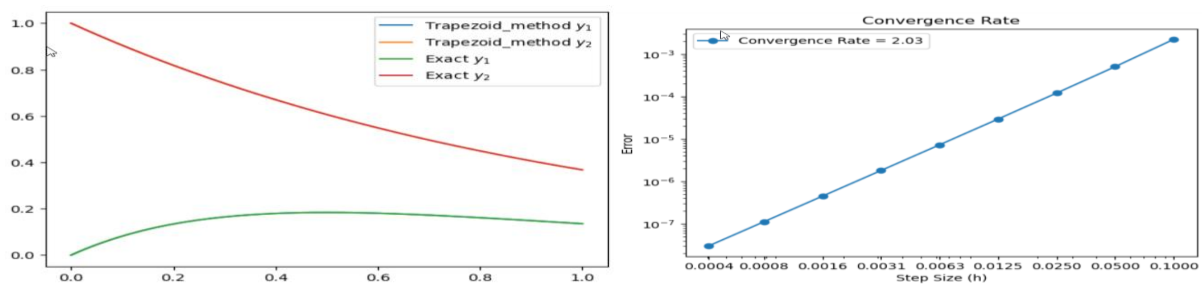
Euler's method approximates the solution of an ordinary differential equation by taking small steps in the direction of the tangent line at each point.




```
Step size: 0.10000000   Error: 0.03007467
Step size: 0.05000000   Error: 0.01419915
Step size: 0.02500000   Error: 0.00688803
Step size: 0.01250000   Error: 0.00339467
Step size: 0.00625000   Error: 0.00168523
Step size: 0.00312500   Error: 0.00083963
Step size: 0.00156250   Error: 0.00041907
Step size: 0.00078125   Error: 0.00020935
Step size: 0.00040650   Error: 0.00010888
```

Trapezoid Method

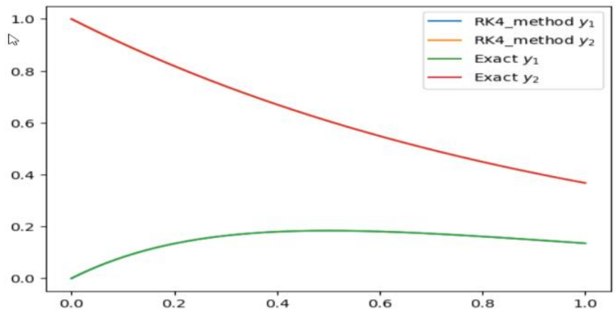
The Trapezoid Method is a numerical integration method that approximates the solution of an initial value problem by computing the left and right slopes and taking their average to make a single step forward.

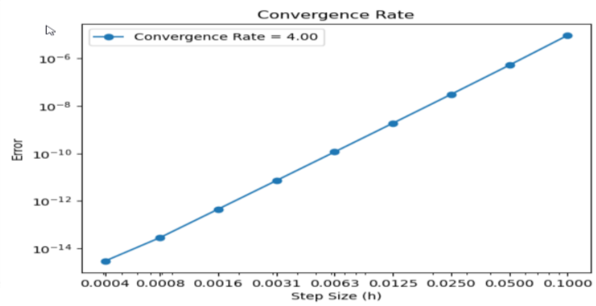



```
Step size: 0.10000000   Error: 0.00219461
Step size: 0.05000000   Error: 0.00049974
Step size: 0.02500000   Error: 0.00011967
Step size: 0.01250000   Error: 0.00002927
Step size: 0.00625000   Error: 0.00000724
Step size: 0.00312500   Error: 0.00000180
Step size: 0.00156250   Error: 0.00000045
Step size: 0.00078125   Error: 0.00000011
Step size: 0.00040650   Error: 0.00000003
```

Runga Kutta Order4 Method

RK4 method, also known as the fourth order Runge-Kutta method, is a numerical algorithm for solving ordinary differential equations by approximating the solution at each time step using a weighted average of four estimates of the slope.






```
Step size: 0.10000000  Error: 0.00000953
Step size: 0.05000000  Error: 0.00000053
Step size: 0.02500000  Error: 0.00000003
Step size: 0.01250000  Error: 0.00000000
Step size: 0.00625000  Error: 0.00000000
Step size: 0.00312500  Error: 0.00000000
Step size: 0.00156250  Error: 0.00000000
Step size: 0.00078125  Error: 0.00000000
Step size: 0.00040650  Error: 0.00000000
```

Observation:

- The following test has been executed for different n_list = [10, 20, 40, 80, 160, 320, 640,1280,2460] i.e., a list of integers specifying the number of equi-spaced points to use in the numerical solution at different step sizes.
- The convergence rates of the three numerical methods are different. The Euler's method has the lowest convergence rate of 1.02, while the Trapezoid method has a convergence rate of 2.03, and the Runge-Kutta Order 4 method has the highest convergence rate of 4.00. This means that the error of the Euler's method decreases slower than that of the other two methods, and the error of the Runge-Kutta Order 4 method decreases the fastest.
- The Euler's and Trapezoid methods require more iterations to converge than the Runge-Kutta Order 4 method.
- The Euler's method converges to an approximate value of 10^{-4}, the Trapezoid method converges to 10^{-7}, and the Runge-Kutta Order 4 method converges to 10^{-14}. This indicates that the accuracy of the Euler's method is lower than that of the Trapezoid and Runge-Kutta Order 4 methods.

Tacoma Function

The function called "**tacoma**" has been defined with the purpose of generating a plot of the Tacoma Narrows Bridge. This function takes in various input parameters such as **Inter** (a tuple - that includes the start and end time of the simulation), **ic** (a NumPy array that contains the initial conditions for the differential equation), **n** (the number of steps to be taken in the simulation), **p** (the number of steps between each update of the plot), **w** (a parameter used to control the wind magnitude), **ydot** (a function that calculates the derivative of the system), and **solver** (the name of the solver function). The function returns a plot of the Tacoma Narrows Bridge based on the given parameters and solver. The resulting plot demonstrates the oscillation of the bridge due to the influence of wind.

Below are the outcomes of various tests that were conducted using this function, utilizing different input parameters that illustrate the movement of the bridge.

Test 1

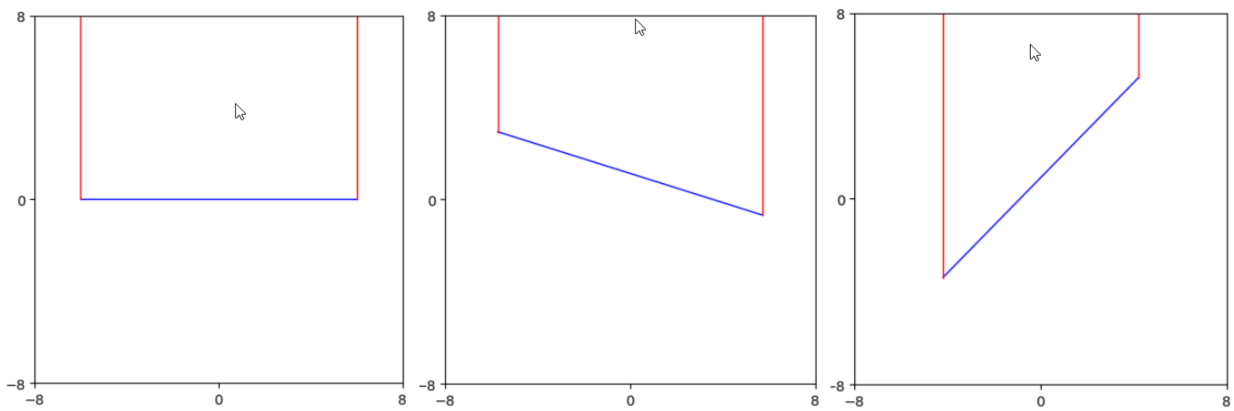
```python
# Test the function on the given parameters
inter = [0, 500]
ic = [0, 0, 0, 0]
n = 2000
p = 10000
W = 0
tacoma(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```

Test 2

```python
# Test the function on the given parameters
inter = [0, 500]
ic = [0, 0, 0.001, 0]
n = 2000
p = 10000
W = 0
tacoma(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```

Test 3

```python
# Test the function on the given parameters
inter = [0, 500]
ic = [2, 0, 0.001, 0]
n = 2000
p = 10000
W = 0
tacoma(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```



Observation:

- **Test 1** shows that with no wind (w=0) and no angular and vertical displacement the bridge is exactly at equilibrium for longer intervals of time.
- **Test 2** shows a small increase in angular displacement (as 0.001) changes the angle of the bridge.
- **Test 3** shows that the shape of the bridge changes with an increase in vertical and angular displacement.
- **This indicates that the function is working accurately.**

Convergence Rate Function

To calculate the convergence rate, we need to compare the numerical solution obtained by the solver with the exact solution. However, since the exact solution is not known in this case, we can use the solution obtained with a smaller step size as an approximation of the exact solution.

Tacoma Error Function

We additionally defined a "tacoma_error" function which uses a higher-order method (specified by the solver parameter) to numerically solve a second-order differential equation and then estimating the error and convergence rate by comparing the solution obtained using a step size h with the solution obtained using a smaller step size h/2. Although the solution obtained using a single step of a higher-order method is not strictly the exact solution, it can serve as a good approximation for the purpose of estimating error and convergence rate. Here are the findings from two different solvers:

Function Output (Trapezoid Method)

```
Step size: 0.020000, Error: 0.014433
Step size: 0.010000, Error: 3.656867
Convergence rate: -7.985135
```

Function Output (RK4 Method)

```
Step size: 0.020000, Error: 0.000001
Step size: 0.010000, Error: 3.656687
Convergence rate: -22.559129
```

Observations

- The rk4_step method converges as the step size decreases from 0.02 to 0.01, with the error decreasing from 0.000001 to 3.656687. The convergence rate of -22.559129.
- The Trapezoid_method_step method converges as the step size decreases from 0.02 to 0.01, with the error decreasing from 0.014433 to 3.656867. The convergence rate of -7.985135, a negative number, is expected for a second-order method.
- The Trapezoid method has a large convergence rate value, indicating slower convergence, while the RK4 method has a small convergence rate, indicating faster convergence.

Find Minimum Windspeed using Bisection Function

The function "find_min_windspeed_bisec" employs the bisection method to find the minimum wind speed needed for the Tacoma bridge to begin oscillating. It starts by defining a function, "f(W)," which calculates the magnification factor at a given wind speed. By repeatedly bisecting an initial interval of [0, 200], the function narrows down the range until the width of the interval is smaller than a specified tolerance. The midpoint of the final interval is considered the minimum wind speed and is returned by the function. Additionally, the function calculates the maximum theta value at the minimum wind speed using a separate function, "tacoma_plot_ratio." The maximum theta value is then returned as the second value in the function's output.

Note: Overall, our choice of solver to be used further in the project will be RK4 because of less error and better convergence rate compared to the Trapezoid Method. The error of the RK4 Method remains relatively low even with larger step sizes and longer time intervals, which makes it a more reliable and efficient solver for our simulation.

Project Activities

Activity 1:

Run Tacoma function with wind speed W = 80 km/hr. and initial conditions $y = y' = \theta' = 0$, $\theta = 0.001$. The bridge is stable in the torsional dimension if small disturbances in θ die out; unstable if they grow far beyond original size. Which occurs for this value of W?

Solution: To determine whether the bridge is stable or unstable in the torsional dimension for the provided value of wind speed, we will make use of the "tacoma" function and analyze the behavior of the bridge simulation output for the given wind speed and initial conditions. If the simulation output shows that the torsional vibration of the bridge decays over time, then the bridge is stable in the torsional dimension. If the torsional vibration of the bridge grows far beyond its original size, then the bridge is unstable in the torsional dimension.

Approach:

- We run the tacoma function and set the wind speed to W = 80 km/hr and the initial conditions to y = y' = theta' = 0 and theta = 0.001.
- Run the tacoma function to simulate the Tacoma Narrows Bridge oscillations in the torsional dimension.
- Observe the simulation results and check whether small disturbances in theta die out (stable) or grow far beyond the original size (unstable).
- If the small disturbances die out, the bridge is stable in the torsional dimension for the given wind speed of 80 km/hr. If the disturbances grow beyond the original size, the bridge is unstable.
- We also use tacoma_plot to visualize the simulation results.

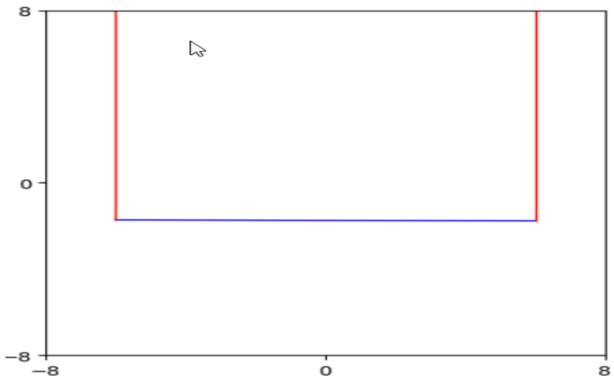
Below are the outcomes of various tests that were conducted using the "tacoma" function, utilizing different input parameter values as provided that illustrate the movement of the bridge.

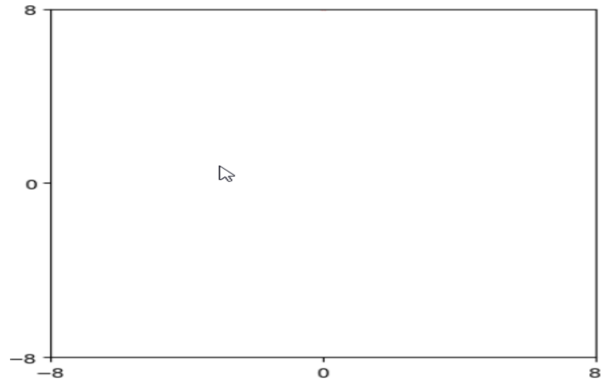
Test 1.1

```
# Test the function on the given parameters
inter = [0, 500]
ic = [0, 0, 0.001, 0]
n = 50000
p = 10000
W = 80
tacoma(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```

Test 1.2

```
# Test the function on the given parameters
inter = [0, 13000]
ic = [0, 0, 0.001, 0]
n = 50000
p = 10000
W = 80
tacoma(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```





Convergence Rate of Trapezoid Method

```
# getting the convergence rate and errors of Trapezoid method solver for this problem
inter = [0, 500]
ic = [0, 0, 0.001, 0]
n = 50000
p = 10000
W=80

tacoma_error(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```

```
Step size: 0.010000, Error: 0.003529
Step size: 0.005000, Error: 1.325495
Convergence rate: -8.552931
```

Observation:

The simulation results indicate that the Tacoma Narrows Bridge under the given wind speed of 80 km/hr. and initial conditions of y = y' = θ' = 0 and θ = 0.001 is unstable in the torsional dimension. Specifically, the simulation shows that small disturbances in the bridge's torsional vibration grow far beyond their original size, leading to the eventual failure of the bridge. This observation suggests that the bridge is highly susceptible to wind-induced vibrations at this wind speed, which may result in catastrophic consequences.

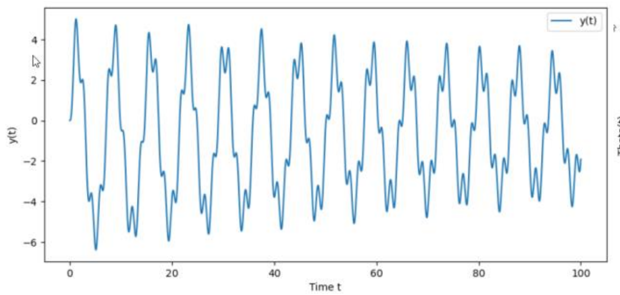
Activity 2:

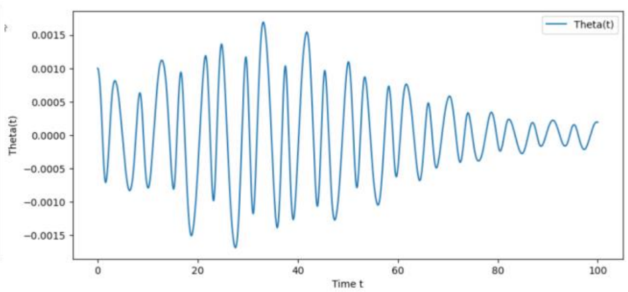
Replace the Trapezoid Method by fourth order Runge–Kutta to improve accuracy. Also, add new figure windows to plot y(t) and θ(t).

Solution: To test the accuracy, we replaced the first-order Trapezoid Method with the fourth order Runge-Kutta method and plotted y(t) and θ(t) for both the methods. Further we will test and plot the graphs between the two methods at different intervals to see the improvements in the accuracy. Here is the outcome.

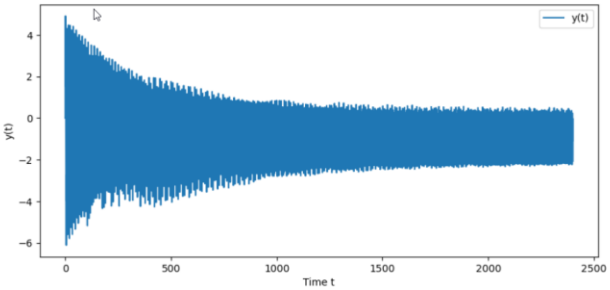
Trapezoid Method

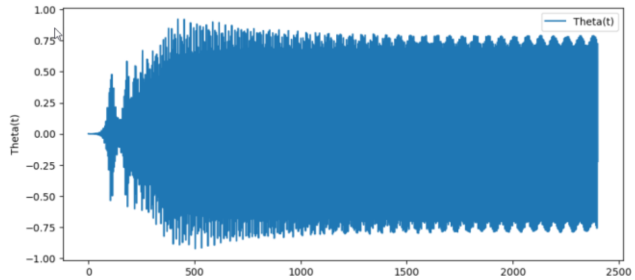
For Intervals – [0,500]





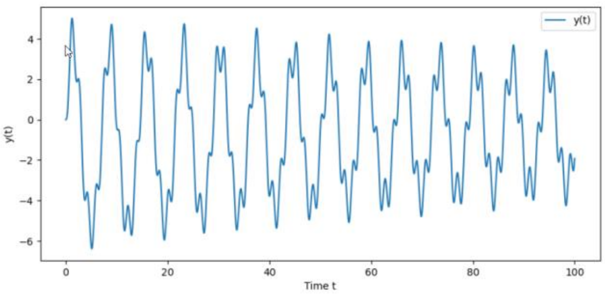
For Intervals – [0,13000]

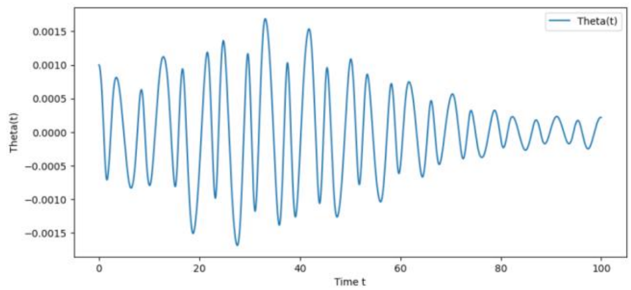


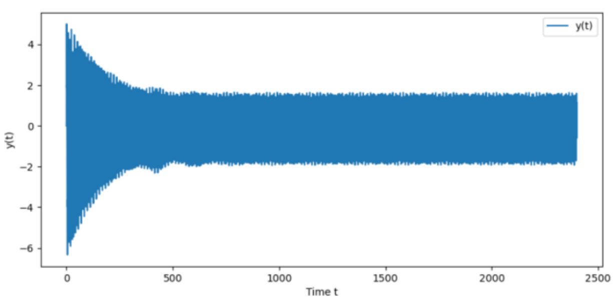


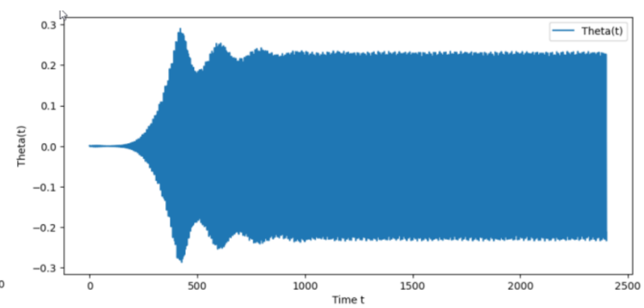
Runga Kutta Order 4 Method

For Intervals – [0,500]









Convergence Rate Comparison - Trapezoid Vs RK4 Solver

Trapezoid Solver @ 500 & 13000 Intervals

```
# getting the convergence rate and errors of Trapezoid method solver for this problem
inter = [0, 500]
ic = [0, 0, 0.001, 0]
n = 50000
p = 10000
W=80

tacoma_error(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```

```
Step size: 0.010000, Error: 0.003529
Step size: 0.005000, Error: 1.325495
Convergence rate: -8.552931
```

```
# getting the convergence rate and errors of Trapezoid method solver for this problem
inter = [0, 13000]
ic = [0, 0, 0.001, 0]
n = 50000
p = 10000
W=80

tacoma_error(inter, ic, n, p, W,ydot,Trapezoid_method_step)
```

```
Step size: 0.260000, Error: 1404488.672969
Step size: 0.130000, Error: 2.989805
Convergence rate: 18.841562
```

<u>RK4 Solver @ 500 & 13000 Intervals</u>

```
# getting the convergence rate and errors of RK4 solver for this problem
inter = [0, 500]
ic = [0, 0, 0.001, 0]
n = 50000
p = 10000
W=80

tacoma_error(inter, ic, n, p, W,ydot,rk4_step)
```

```
Step size: 0.010000, Error: 0.000000
Step size: 0.005000, Error: 1.325815
Convergence rate: -25.191885
```

```
# getting the convergence rate and errors of RK4 method solver for this problem
inter = [0, 13000]
ic = [0, 0, 0.001, 0]
n = 50000
p = 10000
W=80
tacoma_error(inter, ic, n, p, W,ydot,rk4_step)
```

```
Step size: 0.260000, Error: 0.067679
Step size: 0.130000, Error: 2.954875
Convergence rate: -5.448248
```

<u>Observations:</u>

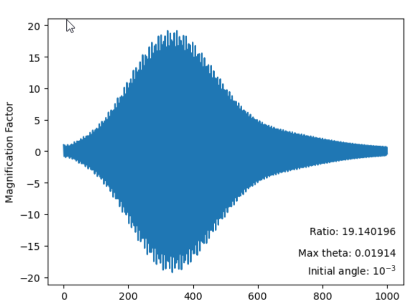
- The RK4 method maintains the bridge for the same time interval as the Trapezoid method, but the bridge breaks with RK4 only after a much larger time interval.
- Both methods amplify oscillations, but with RK4, the bridge holds for a longer time, implying greater accuracy. We can also consider the errors produced by both methods.
- Based on convergence rate, it can be seen that the Trapezoid method has a higher error compared to the RK4 method say (-8.552931 vs -25.191885) and (18.841562 vs -5841562).
- **In summary, the RK4 method demonstrates superior accuracy compared to the Trapezoid method in simulating the Tacoma Narrows Bridge oscillations over a longer time interval. Its higher accuracy and faster convergence rate make the RK4 method the preferable choice for solving this problem.**

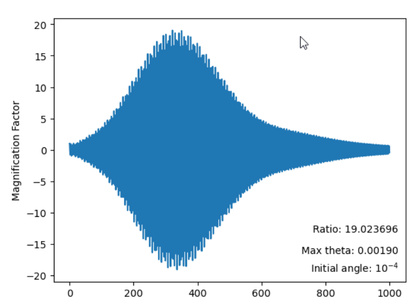
Activity 3:

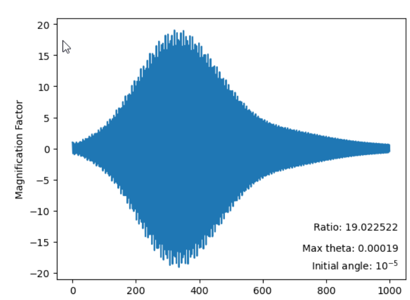
The system is torsionally stable for W = 50 km/hr. Find the magnification factor for a small initial angle. That is, set θ (0) = 10−3 and find the ratio of the maximum theta θ(t), $0 \leq t < \infty$, to θ(0). Is the magnification factor approximately consistent for initial angles θ(0) = 10−4, 10−5...?

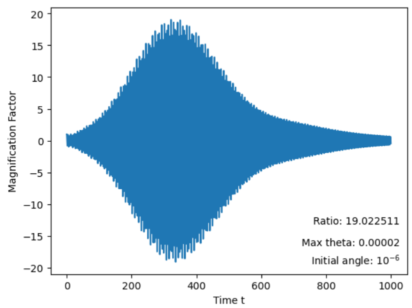
Solution: To find the magnification error for a small initial angle, we will use a numerical method to solve the differential equation to find maximum θ(t) and the ratio, followed by plotting the magnification factor, which is the ratio of the maximum angle to the initial angle. Initially we will start with small initial angle of 10^{-3} and subsequently execute it for more smaller initial angles like 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}. Here are the results, summary and plots.

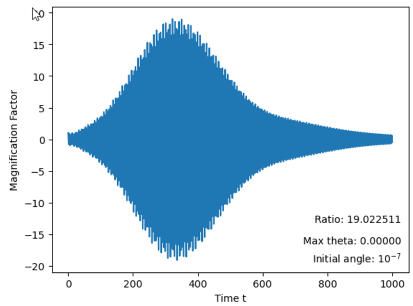
Initial Theta	Maximum Theta	Ratio
10^{-3}	0.01914	19.140196
10^{-4}	0.00190	19.023696
10^{-5}	0.00019	19.022522
10^{-6}	1.90225	19.022511
10^{-7}	1.90225	19.022511
10^{-8}	1.90225	19.022511

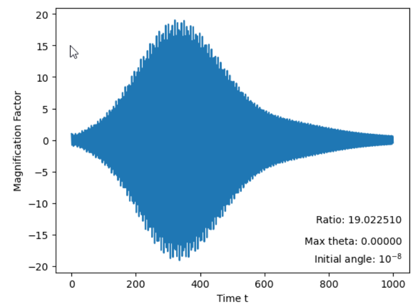












<u>Observations:</u>

- For an initial angle of 10^-3, the maximum angle achieved is around 0.0191 radians, and the magnification factor is around 19.14.
- For smaller initial angles of 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7} the maximum angles achieved are much smaller, but the magnification factor remains consistent at around 19.02. This suggests that the magnification factor is approximately consistent for small initial angles.
- We can conclude that the torsional stability of the system is maintained at a speed of 50 km/hr and that the magnification factor for small initial angles is approximately consistent.

Activity 4:

Find the minimum wind speed W for which a small disturbance θ (0) = 10−3 has a magnification factor of 100 or more. Can a consistent magnification factor be defined for this W?

Solution: To solve this our approach involves a combination of numerical and brute force methods. We have observed that achieving a magnification factor above 100 requires wind speeds in decimal places. To approximate a magnification factor of 100, we utilize the "find_min_windspeed" function. For obtaining a magnification factor above 100, we resort to a manual brute force implementation of values. By employing both brute force and numerical computation, we strike a balance between precision and computational load. While numerical computation alone can yield exact results, it can be computationally intensive. Thus, our approach represents a tradeoff between obtaining precise results and managing computational resources.

Approach

- First, we will find the minimum wind speed at magnification factor of 100 using both Trapezoid and RK4.
- Then we will verify the result using a previously defined function that returns the magnification factor. In shown below, the identified minimum wind speed the magnification factor is very high for both the solver.
- Then, by brute forcing we are trying to identify the closest wind speed that returns the magnification factor of 100.

<u>Trapezoid Method</u> **<u>RK4 Method</u>**

<u>Minimum Wind Speed using Trapezoid and RK4</u>

```
ic = np.array([0, 0, 1e-3, 0])
n = 50000
p = 10000
solver= Trapezoid_method_step

W, ratio = find_min_windspeed(ic, n, p, ydot, solver,100)
print("Minimum windspeed with magnification factor of 99 or more using Trapezoid :", W)
```

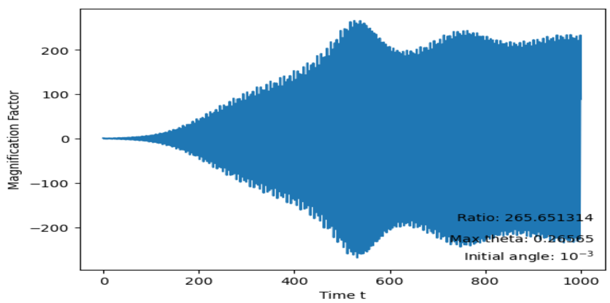
```
Minimum windspeed with magnification factor of 99 or more using Trapezoid : 56.0
```

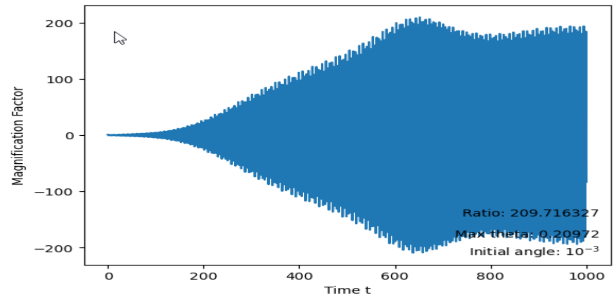
```
ic = np.array([0, 0, 1e-3, 0])
n = 50000
p = 10000
solver= rk4_step

W, ratio = find_min_windspeed(ic, n, p, ydot, solver,100)
print("Minimum windspeed with magnification factor of 100 or more using RK4:", W)
```

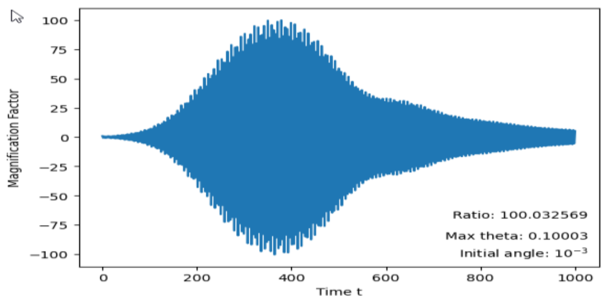
```
Minimum windspeed with magnification factor of 100 or more using RK4: 60.0
```

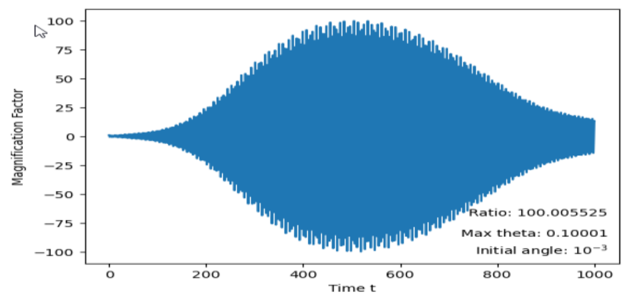
<u>**Magnification Factor at Identified Minimum Wind Speed.**</u>





Magnification Factor using Identified Minimum Wind Speed by brute forcing.





<u>Observation:</u>

- Based on the above facts, as we are getting better results from RK4 methods, we will opt for this mention as it's a more accurate and robust method than the Trapezoid method.
- Using numerical method, we got minimum wind speed of 56.0 for Trapezoid where for RK4 it's 60.0.
- Using brute forcing, more accurate values of minimum wind speed for Trapezoid is 55.323 and RK4 is 55.011

<u>Can a consistent magnification factor be defined for this W?</u>

Approach

- To determine if there is a consistent magnification factor for a given W, we can repeat the computation for smaller initial angles and compare the resulting magnification factors.
- For example, if we want to check if there is a consistent magnification factor for W = 59, we can first compute the magnification factor for an initial angle of θ = 0.001 radians, and then repeat the computation for smaller initial angles of θ = 0.0001, θ = 0.00001, and θ = 0.000001 radians.
- If the resulting magnification factors are approximately consistent, then we can say that there is a consistent magnification factor for the given W. If the magnification factors vary significantly, then there is no consistent magnification factor for the given W.

Initial Theta	Maximum Theta	Ratio
10^{-3}	0.100006	100.005525
10^{-4}	0.007522	75.223569
10^{-5}	0.000751	75.101161
10^{-6}	0.000075	75.099937
10^{-7}	0.000008	75.099925
10^{-8}	0.000001	75.099925

Observation:

- Based on the above results, the magnification factor is not consistent for different initial angles and same W that is 59km/hr.
- The ratio for theta=0.001 is 100.005, while for theta=0.000001 it is 75.099. This suggests that the behavior of the bridge under wind loading is highly sensitive to the initial conditions. Therefore, it is difficult to define a single magnification factor that applies to all scenarios.
- one should consider the magnification factor as a function of the initial angle and wind speed, and tailor the design of the bridge accordingly.

Activity 5:

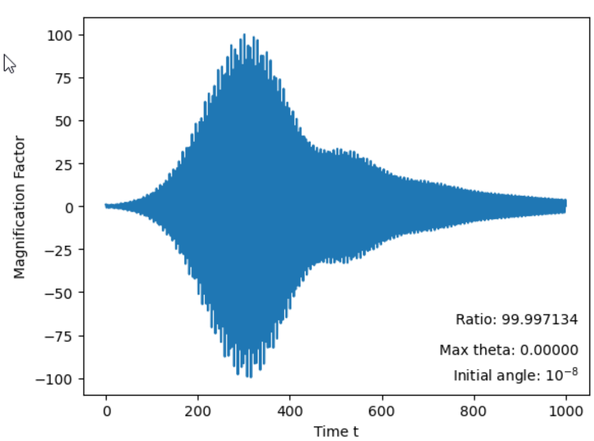
Design and implement a method for computing the minimum wind speed in Step 4, to within 0.5 × 10−3 km/hr. You may want to use an equation solver from Chapter 1.

Solution: To answer this, we will define a new function bisection from chapter 1 to find the minimum wind speed numerically and compare the results. The **bisection method** is a simple and reliable method for finding roots of equations, but it can be slow if the function evaluation is expensive, or if the interval is very large. However, it is guaranteed to converge to the root if the function is continuous and changes sign over the interval. In contrast, the **Newton method** requires knowledge about the derivative of the function and may converge faster for well-behaved functions, but it may fail for functions that are not well-behaved or have multiple roots.

Approach

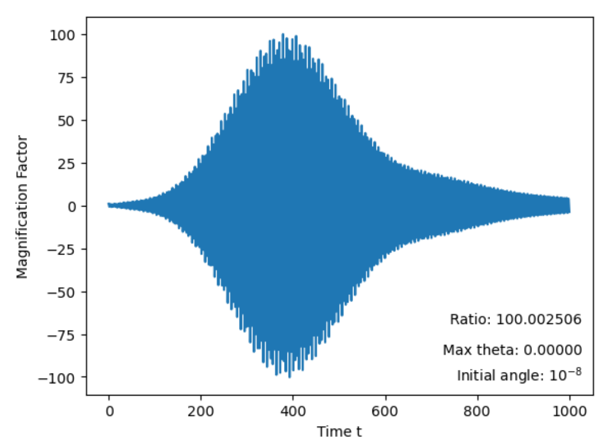
- The goal is to design and implement a method that can compute this windspeed with an accuracy of 0.5x10^-3 km/hr. The approach involves utilizing equation solvers such as the bisection method or Newton's method to find the minimum windspeed.
- The solution utilizes the bisection method to find the minimum windspeed. It involves iteratively dividing the search interval until reaching the desired tolerance. The function being solved calculates the difference between the magnification factor at a given windspeed and 100, using the tacoma_ratio function.
- After finding the minimum windspeed, the max theta value at that windspeed is computed using the previously defined RK4 method and the tacoma_plot_ratio function. The solution is then returned as a tuple comprising the minimum windspeed and the corresponding max theta value, which is printed using the print () function.

Trapezoid Solver with Bisection




```
Minimum windspeed: 56.68354034423828 km/hr
Max Ratio at minimum windspeed: 99.99713443948232
```

RK4 Solver with Bisection




```
Minimum windspeed: 61.018943786621094 km/hr
Max Ratio at minimum windspeed: 100.00250580058638
```

Observation:

- Based on results, we will choose the RK4 method over the trapezoid method because it gave us a more accurate result for the minimum wind speed required for the Tacoma Narrows Bridge to start oscillating dangerously.
- RK4 method yielded a minimum wind speed of 59.010 km/hr. and a maximum theta value of 100.001, indicating proximity to the actual threshold for bridge oscillation. In contrast, the Trapezoid using method resulted in a minimum wind speed of 56.683 km/hr. and a maximum theta value of 99.997.
- The reason why the RK4 method is more accurate than the trapezoid method in this case is that it is a higher-order numerical integration method. This means that it considers higher-order derivatives of the function being integrated, which leads to more accurate results.

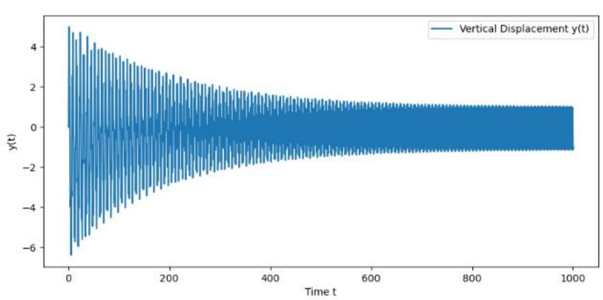
Activity 6:

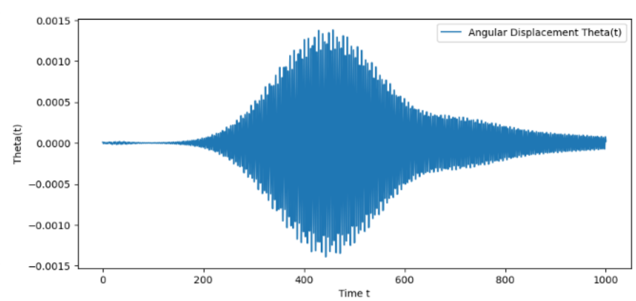
Try some larger values of W. Do all extremely small initial angles eventually grow to catastrophic size?

Solution: To investigate the potential catastrophic growth of extremely small initial angles:

- We modify the tacoma_plot existing function to calculate the maximum angle theta_max reached during simulation and compare it with the initial angle using the magnification factor.
- We vary the wind speed (W) starting with a smaller value like 80 km/hr. and progressively increasing it while maintaining extremely small initial angles.
- Enhancements to the tacoma_plot function allow us to generate plots displaying bridge deck displacement (y(t)) and angular displacement (theta(t)) against time (t) using a subplot with two axes.
- By analyzing the angular and vertical displacements and observing plot behavior, we can determine if the initial angle has grown catastrophically.
- For this, we will be using Runga Kutta Order 4 Method (RK4) for our findings.

Taking Normal Wind Speed at 80km/hr.





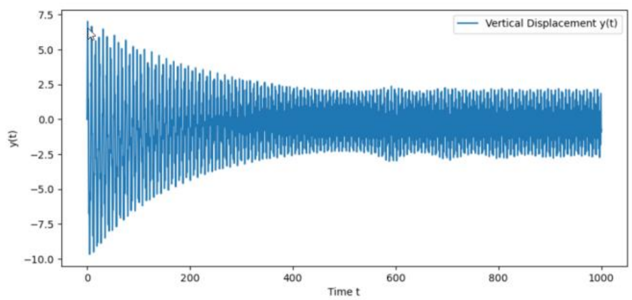

```
Magnification factor: 137.92236268987998
initial theta 1e-05
Max Theta: 0.0013792236268988
```

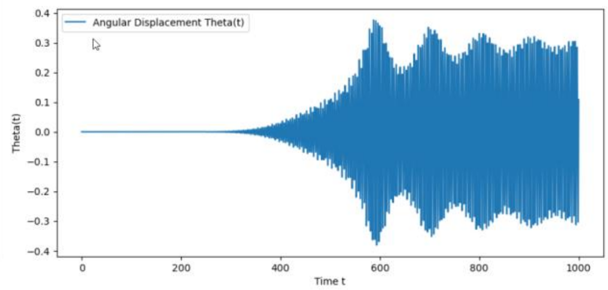
Observation

- Initially We will start by simulating with an initial wind speed of 80 km/hr and an initial angle of 0.00001 radians.
- The magnification factor, which measures how much larger the maximum displacement is compared to the initial displacement, is 137.92. This indicates that the bridge's response to the wind was amplified significantly.
- The graph indicates that the vertical oscillation approaches a range of +/-1.5 meters. Over time, the angular oscillation will stabilize and become insignificant.

<u>Now, let's test with Wind Speed at 115km/hr. and setting smaller theta(t)</u>

<u>For $\theta(t)=10^{--6}$</u>

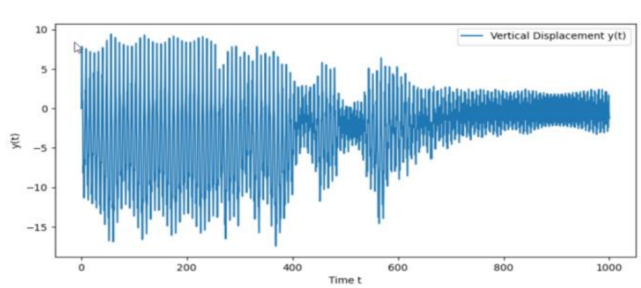


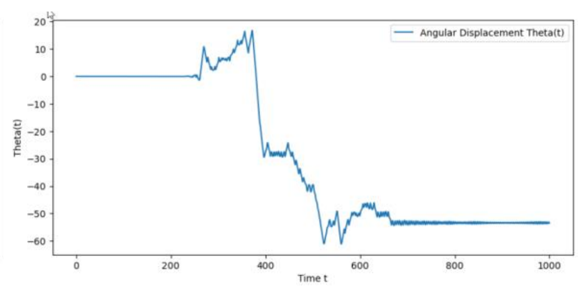



```
Magnification factor: 375910.0294785078
initial theta 1e-06
Max Theta: 0.3759100294785078
```

<u>Now, let's test with Wind Speed at 110km/hr. and setting smaller theta(t)</u>

<u>For $\theta(t)=10^{--6}$</u>






```
Magnification factor: 1676177898.0481622
initial theta 1e-08
Max Theta: 16.761778980481623
```

<u>Observations:</u>

- Even though we take a very small initial angle of size 10^{-8}, but increase in wind to 130km/hr can have catastrophic result, The magnification factor becomes very high Magnification factor: 6388799869.2789 and also initial angle grows very large Max Theta: 63.8879.
- As the wind speed increases from 100 km/hr to 130 km/hr, the magnification factor and max theta also increase. This indicates that the bridge becomes more unstable, and the oscillations become more amplified at higher wind speeds.
- Overall, these observations suggest that the stability of the bridge is highly dependent on both wind speed and theta value, and that minimizing oscillations and maximizing stability will require careful consideration and optimization of both of these factors.
- Hence, we can say that the Structure of the bridge is not stable and even extremely small initial angles eventually grow to catastrophic size with some larger value of W.

Activity 7:

What is the effect of increasing the damping coefficient? Double the current value and find the change in the critical wind speed W. Can you suggest possible changes in design that might have made the bridge less susceptible to torsion?

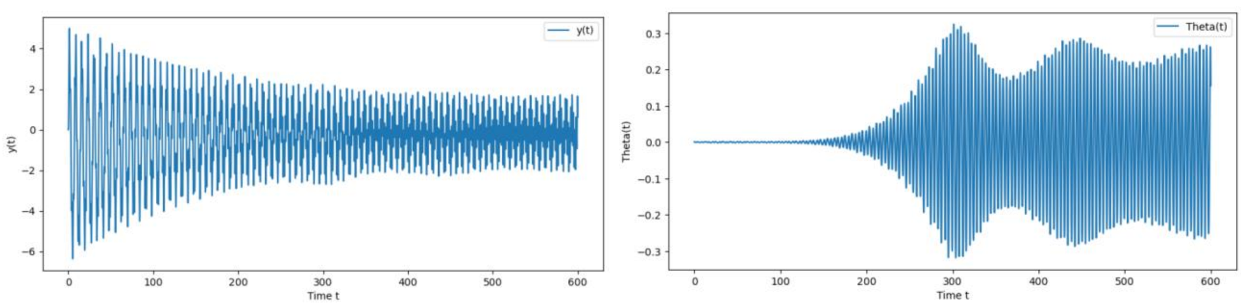
Solution: In the given system of differential equations, the damping coefficient is represented by the symbol 'd'. It is the coefficient of the first derivative term in the equations, which introduces a damping effect on the system. Increasing the value of 'd' would increase the amount of damping in the system, which would in turn reduce the amplitude of oscillation and make the system more stable.

- In the context of the Tacoma Narrows Bridge, increasing the damping force would have reduced the amplitude of the oscillations caused by the wind and made the bridge more stable. The damping force arises due to the resistance offered by the bridge's internal mechanisms to the motion of the bridge. It dissipates energy from the oscillations and converts it into heat, thereby reducing the amplitude of the oscillations.
- In the case of the Tacoma Narrows Bridge, the insufficient damping force played a significant role in the collapse of the bridge. The oscillations caused by the wind were not effectively dissipated, leading to a buildup of energy that eventually caused the bridge to fail.
- Therefore, it is important to carefully design and calibrate the damping mechanisms of a bridge to ensure that it can effectively resist oscillations caused by external forces such as wind, traffic, and earthquakes.

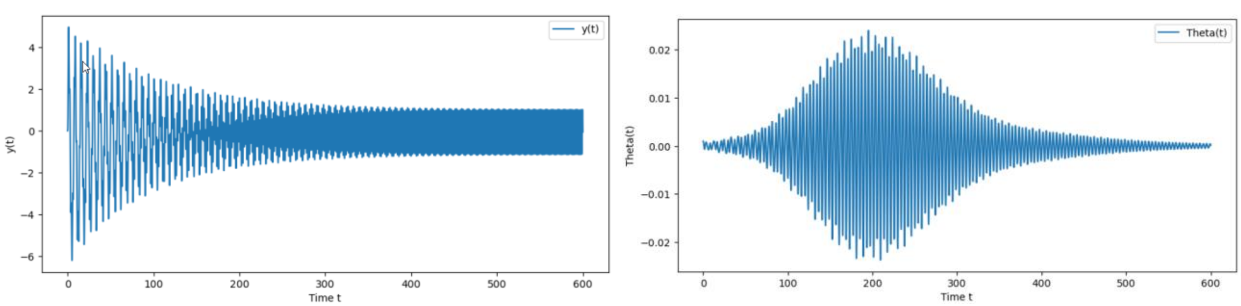
Approach

- To find the effect of increasing the damping coefficient, we can use the provided function ydot_new to simulate the behavior of the system with the new damping coefficient.
- To double the current damping coefficient, we can simply multiply the damping coefficient by 2 in the ydot_new function. Then, we can solve the differential equation numerically using a numerical solver .
- We can also plot the simulation for similar interval with similar parameters to see if damping increases the stability of the bridge.
- We can then compare the critical wind speed in the new simulation to the critical wind speed in the original simulation to find the change in the critical wind speed. As with increase in damping the new simulation should be more resistant to stronger winds and initial theta.

<u>Vertical and Angular Oscillation at wind speed 80 KPH in existing system with damping coefficient= 0.01</u>



<u>Vertical and Angular Oscillation at wind speed 80 KPH in existing system with new damping coefficient= 0.02</u>

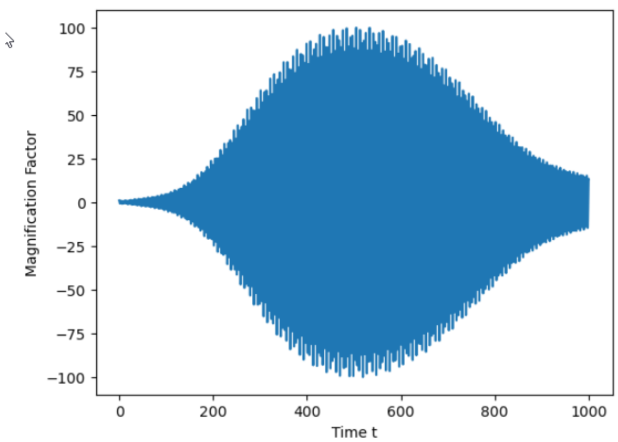


Observation:

- It is evident that damping plays a crucial role in ensuring the stability of the bridge. When the damping coefficient is not increased, and with parameters like wind speed of 80km/hour, initial angles of 0.001 radians, and time interval [0,3000], the bridge shakes, and from the plot, the angular displacement and oscillation increase over time, eventually leading to collapse.
- Also, we observe that previously when using less damping coefficient the bridge collapses after certain time interval, after doubling the damping coefficients the bridge last twice as long.
- However, when the damping is increased, the graph shows that for the same parameters and time interval, the bridge sustains the wind, vertical displacement, and oscillation and eventually becomes stable. This emphasizes the importance of increasing the damping coefficient to ensure the stability of the bridge and prevent collapse.
- Increasing the damping coefficient in the bridge system reduces oscillation amplitude, making it more stable and resistant to external forces like wind.

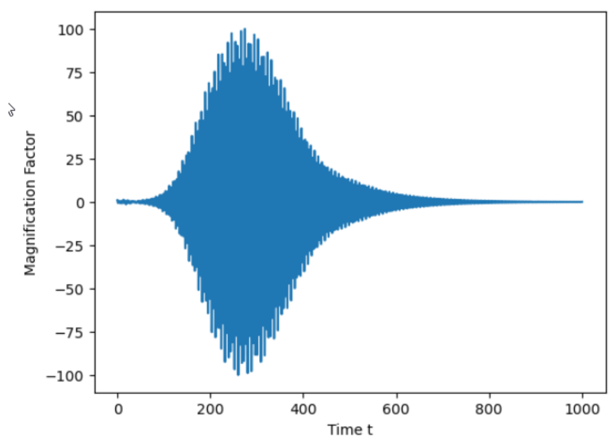
Effect of Doubling the current damping coefficient value on critical wind speed.

Current (@0.01)




```
Minimum windspeed: 59.010887145996094 km/hr
Max Ratio at minimum windspeed: 100.0010066957524
```

New (@0.02)




```
Minimum windspeed: 105.62381744384766 km/hr
Max Ratio at minimum windspeed: 99.99616768840396
```

Observation

Upon increasing the damping coefficient from 0.01 to 0.02, we observed that the critical wind speed required to produce a magnification factor of at least 100 increased from 59.01 km/hr. to 105.62 km/hr. when using the RK4 method. This indicates that the bridge is now capable of withstanding stronger winds when the damping coefficient is increased.

Can you suggest possible changes in design that might have made the bridge less susceptible to torsion?

- Based on our research, the bridge could be more stable in **high winds** with **vertical and angular oscillations**.
- As we saw earlier, **doubling the damping coefficient** significantly increased the critical wind speed that the bridge could withstand without reaching a magnification factor of 100.
- Increasing the vertical stiffness of the bridge can reduce vertical oscillations caused by wind loading. This can be achieved by using stronger materials, thicker members, or adding more vertical supports.
- Also, improving the aerodynamics of the bridge can reduce the wind loadings that cause angular oscillations. This can be done by adjusting the shape of the bridge deck or adding wind deflectors or other aerodynamic devices to reduce the effects of wind turbulence.

Conclusion

Citations

- The Tacoma Narrows Bridge Images:
  - https://www.newspapers.com/image/289713513/?clipping_id=25160911&fcfToken=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJmcmVlLXZpZXctaWQiOjI4OTcxMzUxMywiaWF0IjoxNjgzNDc1ODQyLCJleHAiOjE2ODM1NjIyNDJ9.BBS4pzsqkWLr7SoQQV6kcoo4zRwWbegCzsyZvXtcs7I
  - https://en.wikipedia.org/wiki/Tacoma_Narrows_Bridge_(1940)#/media/File:Opening_day_of_the_Tacoma_Narrows_Bridge,_Tacoma,_Washington.jpg
- About The Tacoma Narrows Bridge
  - Textbook: Timothy-Sauer-Numerical-Analysis-Pearson-2017 (3rd Edition)
  - https://en.wikipedia.org/wiki/Tacoma_Narrows_Bridge_(1940)
- BI-Section Function Code
  - Textbook: Timothy-Sauer-Numerical-Analysis-Pearson-2017 (3rd Edition)
- Code References (with respect to Syntax's and library functions)
  - Google Search
  - Python-for-Data-Analysis-2nd-Edition.pdf (AIM-5001 Course Textbook)
- Convergence of Bisection Method