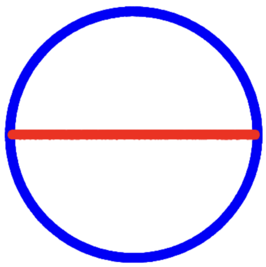
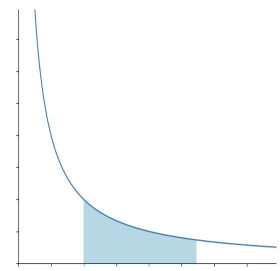


Modeling π and *e*

through approximation and application



Adam Mahmoud & Jack Galetti
Physics 77 - Capstone Report
12 May 2023

Abstract

This project explores numerical, geometric, and statistical methods to approximate the mathematical constants pi (π) and Euler's number *e*. The most accurate of these approximations are then applied to a physics-based simulation, modeling the behavior of a damped pendulum undergoing small angle oscillations. In this application, we observe the decaying amplitude of the pendulum's angle over time, as well as the impact that damping has on the pendulum's arc length within each period. The investigations presented in this project contribute to the understanding of mathematical constant approximations, with specific regards to their relevance in physical models.

Introduction & Motivation

Pi and *e* are two of the most ubiquitous constants that we encounter in all aspects of math and the physical sciences. Having put these numbers to use so often, we were inspired to investigate a wide variety of techniques that can be used to derive these famous constants. In doing so, we set out to provide deeper analytical, geometric, and probabilistic intuition into what π and *e* are, and how they can be applied to real-life scenarios—specifically physics based phenomena. In some of the methods we used, we thought up strategies for approximation that have not been widely applied in preexisting available literature, if at all. This includes the particular way we utilized Monte Carlo simulations to approximate π by the ratio of lengths, rather than areas, as well as the estimate of *e* through precise alteration of the upper bound of integration. Conversely, the other approaches we employed are widely used in approximating π, including observations of the perimeter of an n-gon, infinite summations and products, and Buffon's needle. To wrap up our findings, we were inspired to apply our results to phenomena we learned about in physics classes—that of a pendulum undergoing dampening forces.

Methods/Calculations & Results

We will now delve into the specifics of the calculations and modeling approaches that were taken for each strategy in approximating π and *e.* As expected, this section will include approximations for π from: Monte Carlo, polygon perimeter, Leibniz formula, Wallis product, and Buffon's needle; and for *e* an approximation by integration. We conclude with the physics application of a damped pendulum. We will discuss these strategies in their entirety, including all results from each method, accompanied by any successes or failures encountered along the way.

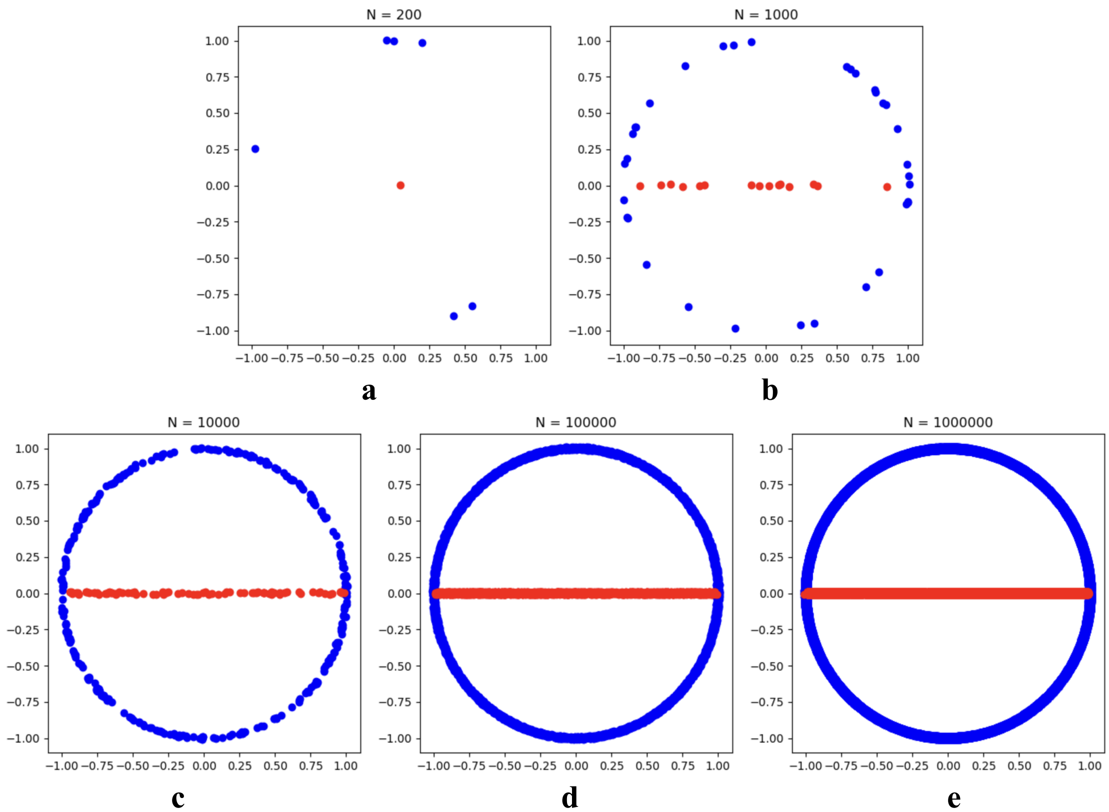
π - Monte Carlo

As a succinct definition, a general Monte Carlo approach "may be thought of as a collection of computational techniques for the (usually approximate) solution of mathematical problems, which make fundamental use of random samples" (Johansen). There is a classic monte carlo approach that is generally taken to get an approximation of π. In this popular method, many points are randomly plotted within a 1x1 square that has a circle with radius 0.5 inscribed into it. The idea here is that as the number of randomly generated points increases, and knowing the formula for the area of a circle to be πr^2, then through simple calculations, π can be approximated as four times the ratio of the number of points that lie within the circle over the number of total points plotted.

Rather than using this common approach that examines the areas of a circle inscribed in a square, we were inspired to utilize the Monte Carlo method in the context of the pure definition of π—that is, the ratio of a circle's circumference to its diameter. In order to simulate this, we first needed to plot points (which were randomized coordinate decimals between -1 and 1 with numerous digits) on the lengths of a circle's circumference and diameter. This is where we ran into an issue, as it is nearly impossible to have one of these random points lie exactly on the circumference of the circle with no numerical tolerance, even with millions of points being plotted. In order to combat this issue, we allowed the points to count as 'landing' on the circle and its diameter so long as they were within 0.01 of the true coordinates defining these curves. With this tolerance, we began seeing points being plotted on both the circle's circumference and diameter at upwards of 200 points, seen in **Figure 1a** below. With all the rules properly in place for this Monte Carlo simulation, all that was left was to gradually increase the total number of randomly plotted points, and compare the ratio of the amount that lay on the circumference to the diameter. Snippets of an animation we created to demonstrate this process are shown below in **Figure 1.**

As expected, the approximation for π steadily grew more accurate as the number of plotted points increased, with an estimation of 3.1695 (true value to 5 decimal places is 3.14159) when 1,000,000 points were used (**Figure 1e**). In the end, we concluded this Monte Carlo approach to be on the same order of efficiency and accuracy as the commonly practiced area method described earlier. And while these random numbers nicely approach the value of pi, there is still much room for improvements in accuracy.

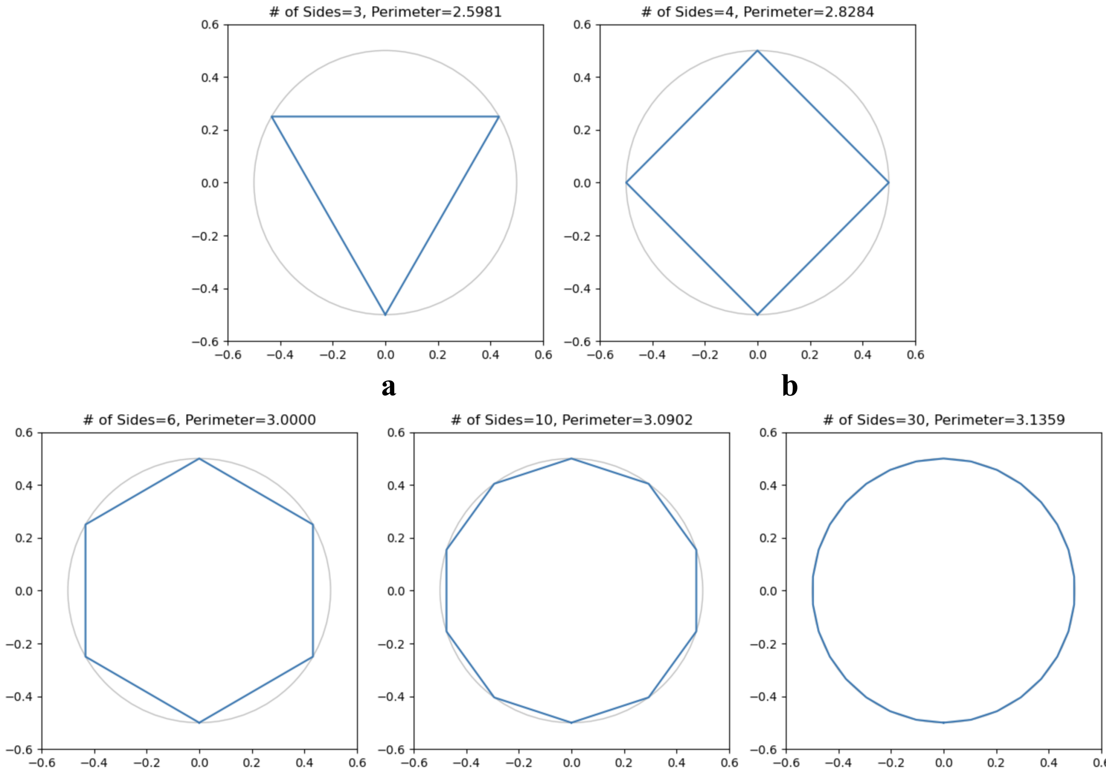
Figure 1



π - Polygon Perimeter

This next approach is a highly intuitive, geometric method of approximating π. Knowing that the circumference of a circle is given by the formula 2πr, we can create a circle of radius 0.5; if we are able to find the circumference of this circle we will immediately know the value of π. In order to approximate the circumference, we choose to inscribe a regular polygon into this circle, as the perimeter of an n-sided polygon (n-gon) is easily calculated as the sum of all the side lengths (or n*l, where n is the number of sides and l is the length of each side of the polygon). As seen below in **Figure 2a**, we start with a three-sided regular polygon (or an equilateral triangle), calculate its perimeter, and then increase the number of sides of the polygon. Specific frames of an animation we coded to visualize this increase in polygon sides are shown in **Figure 2**. As expected, the perimeter of the polygons quickly approach the value of π, and visually become indistinguishable from a true circle. This approximation strategy proves to be much more accurate than the Monte Carlo approach, because when we use a polygon with 1,000,000 sides, the estimation of π only begins to deviate from the true value at the 11th decimal place. In fact, this method turns out to be the most accurate of all five of our approaches, and with its simplistic geometric nature, is a nice and convenient way to approximate π.

Figure 2



c d e

π - Leibniz Formula

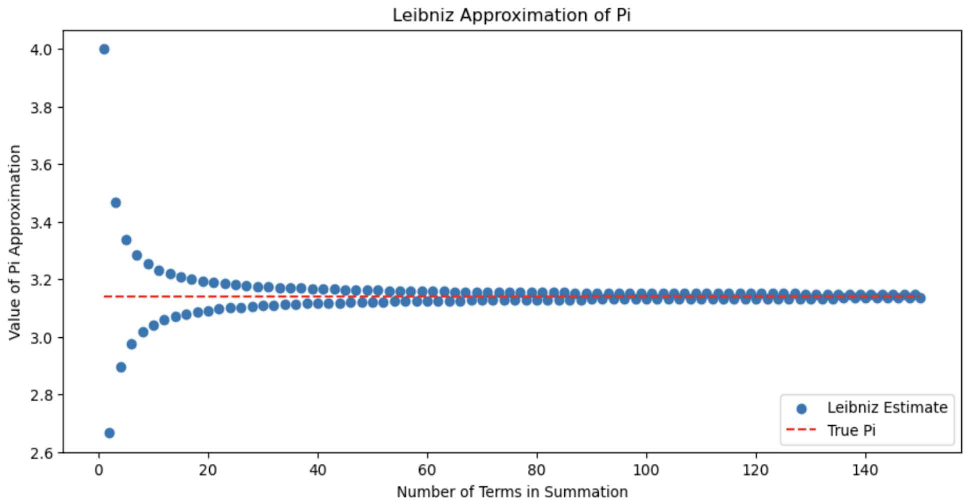
Named after German mathematician Gottfried Leibniz, the Leibniz formula is an infinite series that can be used to approximate π. This formula is derived from the fact that the arctangent of 1 is equivalent to $\frac{\pi}{4}$. The Leibniz formula is expressed in **Eq. 1**.

$$\frac{\pi}{4} = \sum_{i=0}^{\infty} \frac{(-1)^i}{2i+1} \qquad \textbf{(Eq. 1)}$$

The goal with this approximation is to increase the number of terms used in the summation to get closer and closer to the value of π. It is worthwhile to note that due to the fact that this is an alternating series, as each successive term is multiplied by -1, the approximate values will in turn alternate around the true value of π. This pattern can be seen in **Figure 3**, which models Leibniz's formula for approximating π up to 150 terms in the series.

To compare the accuracy of this approach to our others, we again use 1,000,000 terms, in which the Leibniz formula returns the value for π correct to 5 decimal places. Although this method does not appear as accurate as the polygon perimeter approach, the Leibniz formula and similar calculations built off of it are the most common methods that are still being employed to try to find as many digits of π as possible. The reason for this largely surrounds the fact that this formula is a more definition-based calculation, and less ambiguous than a polygon's perimeter.

Figure 3



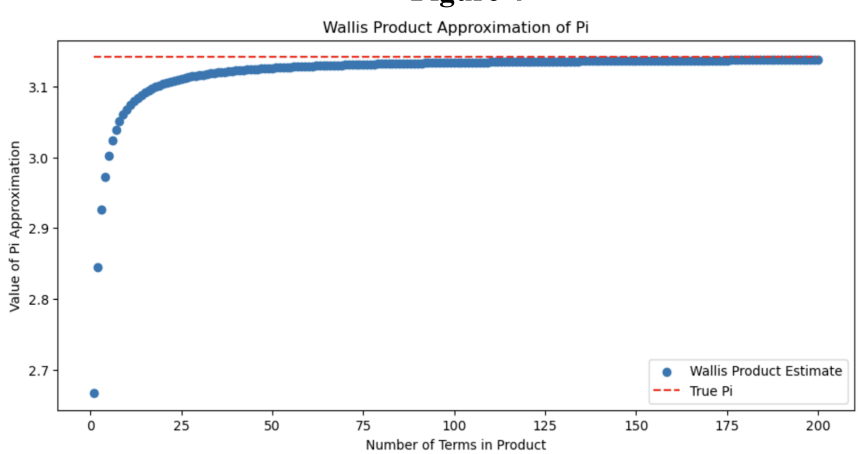
π - Wallis Product

Rather than the infinite series that was the Leibniz formula, The Wallis product (named after mathematician John Wallis) is an infinite product of terms, defined by **Eq. 2**, that results in the value of π. This equation is predominantly derived from integration of the sine function, which of course is largely related to π.

$$\pi = 2 \prod_{i=1}^{\infty} \frac{(2i)^2}{(2i-1)(2i+1)} \qquad \textbf{(Eq. 2)}$$

Just like with the Leibniz formula, the approach we took here was simply to compute the Wallis product for an increasing amount of terms. A key difference to note with this method of calculating π compared to the Leibniz equation is that there is no alternation around the true value of π, as there is no multiplication by a negative. This allows for a much smoother and more efficient approach towards π, which is shown graphically in **Figure 4**.

Figure 4



We again compare the accuracy of this approximation method by using 1,000,000 terms in the Wallis product. Just like with the Leibniz formula, we get an output that is accurate to 5 decimal places of π.

π - Buffon's Needle

This final method we employed to approximate π is one regarding geometric probability. The scenario posed is the following:

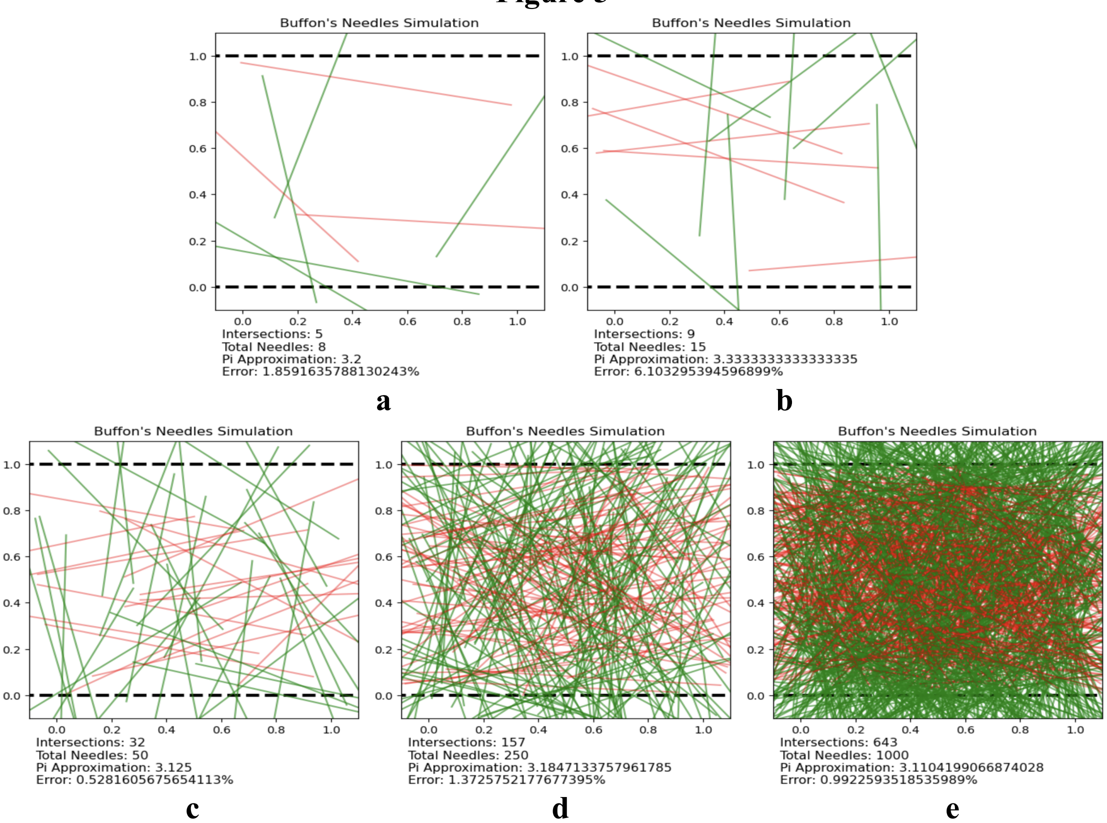
> *Suppose we have a floor made of parallel strips of wood, each of the same width. If we drop a needle on the floor, what is the probability that the needle will intersect a line between two strips?*

Surprisingly, this probability is closely related to the value of π after some elementary calculations, given by **Eq. 3**, where L represents the length of the needles and D is the distance between lines.

$$\pi \simeq \frac{2L}{D} \cdot \frac{\#\ of\ Needles}{\#\ of\ Intersections} \qquad \textbf{Eq. 3}$$

In **Figure 5**, we simulated Buffon's needle with L=D=1. Green needles are ones that pass through the dashed lines, while red needles do not. The data and calculations in each image confirm that as the number of needles increased, the approximation of π became more accurate.

Figure 5



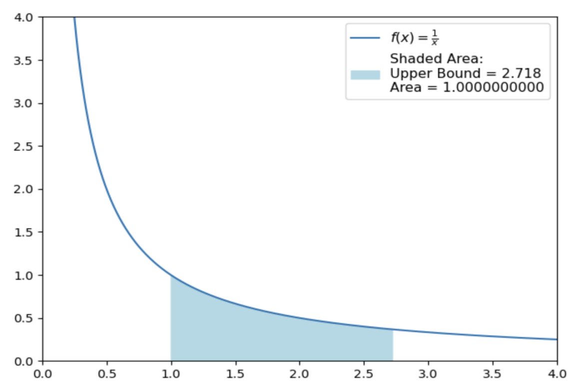
As is routine, we ran the calculations of this simulation for 1,000,000 needles (we did not actually graph this, as this amount of needles would cause the code to crash). Still, even with this many needles, the average trial of the simulation only tends to yield an approximation of π correct to one decimal place. Evidently, this is one of our least precise approximation methods, but it is perceptive to see π emerge from this random setup.

e - Integration

We now move on to Euler's number, *e*. A unique property of *e* is expressed in **Eq. 4**, and is seen graphically in **Figure 6**.

$$\int_1^e \frac{1}{x}dx = 1 \qquad \textbf{Eq. 4}$$

Figure 6



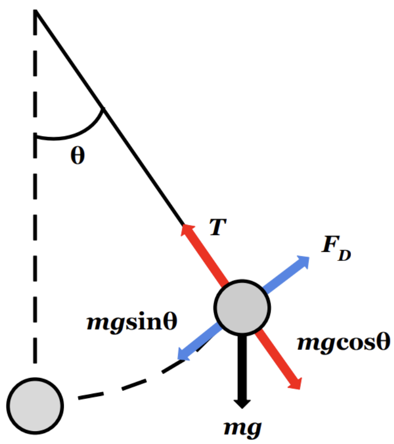
Our goal to approximate *e* is entirely built around this fact. We implemented code that utilizes left-hand, right-hand, and trapezoidal Riemann Sum approximations of integration, using 1,000,000 steps. We then made algorithms to integrate $\frac{1}{x}$ with a lower bound, and an upper bound that varies between 1 and 3. The algorithm would output the upper bound *x* value for each Riemann Sum approximation that results in the integral being equal to 1. As expected, this value was a very close approximation to the known value of *e*. What did come as a surprise, however, was that the right-hand Riemann Sum approximation yielded the most accurate value for *e*, which was correct to 6 decimal places (compared to four decimal places by both the left-hand and trapezoidal approximations).

π and e - Application to Damped Pendulum

Our final modeling goal was to demonstrate these constants in action. We decided to observe a damped pendulum, which is one that is undergoing some type of dampening/drag force, such as air resistance. In general, the angle this pendulum makes (shown in **Figure 7**) can be expressed by the second order, non-linear differential equation given in **Eq. 5**.

$$\frac{d^2\theta}{dt^2} + \frac{b}{m}\frac{d\theta}{dt} + \frac{g}{L}\sin\theta = 0 \qquad \textbf{Eq. 5}$$

Figure 7



In order to work with a linear differential equation, we decided to use small angle approximations for the pendulum, making the *sin(*θ*)* term approximate to simply θ. This allowed us to calculate a general solution to this differential equation, given by **Eq. 6**.

Eq. 6

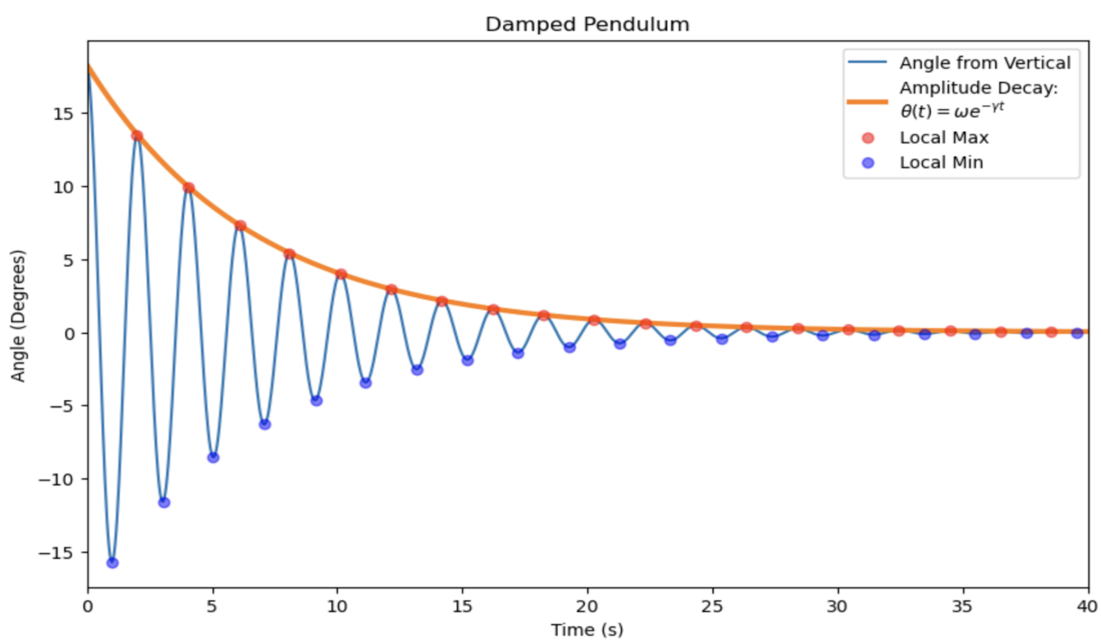
$$\theta(t) = e^{-\gamma t}\left(c_1\cos(\alpha t) + c_2\sin(\alpha t)\right) \text{ where } \alpha = \omega\sqrt{1-\gamma^2},\ \omega = \sqrt{\frac{g}{L}},\ \text{and } \gamma = \frac{b}{2m}$$

Ultimately, our two objectives were to incorporate *e* by modeling the underdamped oscillation of the pendulum's angle, and integrate π by showing the declining arc length subtended by the pendulum with each successive period. Plotting **Eq. 6** helps achieve this first goal, and that graph is shown in **Figure 8**. The decaying amplitude of the pendulum's angle over time is given by **Eq. 7**. However, rather than utilizing the true values of π and *e* in this equation, we incorporated our approximations calculated earlier; that is, the polygon perimeter estimation of π, and the right-hand integration method for *e*.

$$\theta(t) = \sqrt{\frac{L}{g}}\, e^{-\gamma t}\,\frac{180}{\pi} \qquad \textbf{Eq. 7}$$

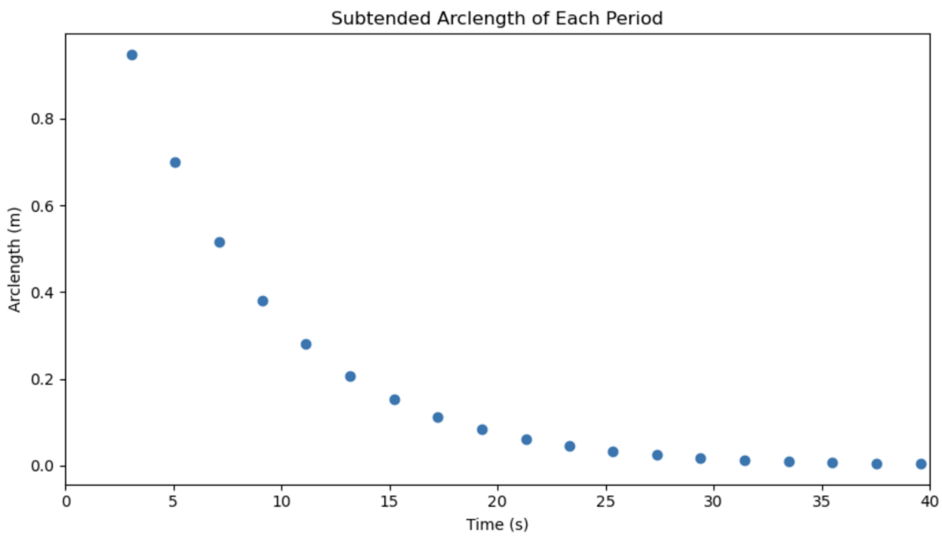
Even with our approximations being employed, we got the fitted graph we were anticipating, as shown in **Figure 8**.

Figure 8



To achieve our second objective, we needed to introduce our approximated π again in order to convert from degrees to radians, a necessary step in calculating the arc length (formula given by θr, where r is the radius). As expected with our relatively precise value of π, the predicted model was not derailed, and we were able to produce a sensible graph of arc length of each period over time, shown in **Figure 9**.

Figure 9



Ultimately, the results of these models fit closely to our expectations, and this application demonstrates the ability and accuracy of approximated results to be used in real-world scenarios.

Conclusion

This investigation into the approximation of the constants π and *e* provided great insight into these numbers, where they come from, their significance, and their potential for application. Through the implementation of a variety of approaches, such as random generation, infinite series, and geometric means, we saw that the polygon perimeter approach returned the most accurate value of π, while right-handed integration resulted in the best approximation for *e*. Our connection to a damped pendulum system evidenced the practical application of these constants, even when they are approximated.

Special thanks to Professor Kolomensky for his time spent outside of class to teach coding animation strategies, and to Mr. Lee for his insightful feedback towards the project.

References

"Buffon's Needle Problem." MapleSoft, www.maplesoft.com

Johansen, A. M. "Monte Carlo Method." Science Direct, 2010, www.sciencedirect.com/topics/medicine-and-dentistry/monte-carlo-method.

"Wallis Product." Wikipedia, 31 Mar. 2023, https://en.wikipedia.org/w/index.php?title=Wallis_product&action=history