Introduction to Calculus

Paul Lutus

http://arachnoid.com

September 1, 2016

Most recent revision: June 11, 2018

Abstract

This article provides an overview and introduction to calculus. It's intended for general readers, nonspecialists, and shows the topic's key concepts in a transparent, approachable way.

The article's purpose is to help readers see that calculus is not only relatively easy to understand, but is a useful and accessible intellectual skill without which some aspects of the the modern world are indecipherable – population growth, space travel, climate change, epidemics and control of diseases, economic policy, investment strategies and others – all of which require a knowledge of how and why things change, the kind of knowledge calculus provides.

Some familiarity with secondary school algebra is desirable to be able to follow most of the article's content. Some new topics, such as functions, are introduced and explained before being used.

The article includes tutorial exercises that show concepts without overwhelming the reader in detail, and covers related topics such as differential equations and computer algebra systems.

Contents

List of Figures

List of Tables

1 Basic Concepts

1.1 Math Mythology

While introducing calculus, this article tries to demolish a pervasive American myth – that calculus is outside the intellectual grasp of everyday people. In contrast to the practice in many other countries, American students are told that calculus is an advanced mathematical topic, one neither suitable for, nor within the mental abilities of, ordinary people. The author believes this is a crippling and misleading belief, one easily falsified by some exposure to the topic itself. In this article, the author strives to explain the mathematical concepts in the simplest, most accessible way.

1.1.1 Stationary Math

The mathematical topics taught before calculus produce static results, unchanging outcomes for provided arguments. Calculus differs by describing dynamic systems, systems that are moving and changing. If we want to mathematically describe a rock we can weigh it, compare its volume to its mass and derive a density, compute a ratio of volume to surface area, things like that – fixed outcomes, no motion or change. But imagine that we throw the rock and try to predict the rock's path. Because the rock is now moving, we need calculus.

1.2 Car Example

The key idea of calculus, on which the entire field stands, is the relationship between a quantity and a *rate of change* in that quantity. For example, we might want to describe the position of a moving car as time passes:
- The car's acceleration, controlled by the accelerator pedal, is a *rate of change* in the car's velocity.
- The car's velocity, recorded by the speedometer, is a *rate of change* in the car's position.
- Expressed in everyday language, position is "where you are", velocity is "how fast you're moving", and acceleration represents "how velocity is changing."
- These three quantities – position, velocity and acceleration – are bound together in such a way that, with the methods of calculus, we can use any one of them to find the other two.
- For example, we can note a car's speedometer (velocity) readings over time and use that to reconstruct the car's position, without any other information.*

1.2.1 Car motion data

Here's an an example. Let's say we're in a car that's stationary at time zero and that begins to accelerate along a road. Here's a table and plot that shows the car's acceleration, velocity and position at different times:

| Time	Acceleration	Velocity	Average Velocity	Position
0	4	0		0
			2	
1	4	4		2
			6	
2	4	8		8
			10	
3	4	12		18
			14	
4	4	16		32
			18	
5	4	20		50
			22	
6	4	24		72
			26	
7	4	28		98
			30	
8	4	32		128
			34	
9	4	36		162
			38	
10	4	40		200

Table 1: Acceleration/Velocity/Position
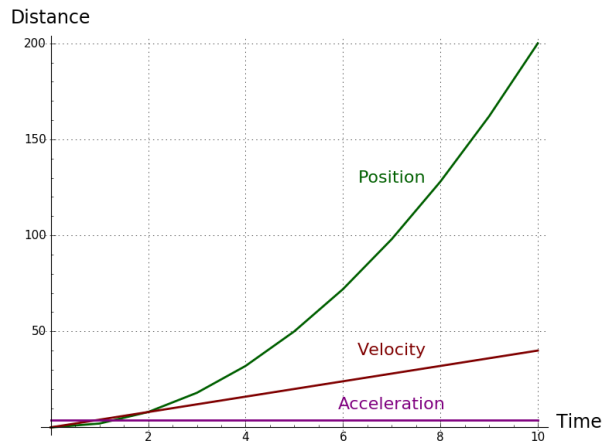
Figure 1: Acceleration/Velocity/Position

(For a technical explanation of this table's design, see the "Data Table Design" appendix on page 32.)

---

*In principle, if the car were equipped with an accelerometer[1], we could use that to reconstruct both velocity and position over time (this is one way astronauts find out where they are).

4

1.2.2   Differentiation and integration

Three quantities are listed in Table 1 – acceleration, velocity and position. Here's what we can do with this information:
- As explained above, using calculus methods, *we can use any one of the values to find the other two*. For example, if we only have a list of accelerations over time, we can use it to compute velocity, and we can use velocity to compute position.
- Also, by performing the opposite or *inverse*[2] operation, and with only a list of the car's positions over time, we can compute the car's velocity and acceleration over time.
- We can do this because velocity is a *rate of change* in position, and acceleration is a *rate of change* in velocity. These values are said to be *interdependent*.

1.2.3   Concepts

Before we move on, here are some important concepts:
- Finding a rate of change in a quantity is called *differentiation*, and the result is called a *derivative*[3].
- The inverse operation, summing rates of change to obtain a quantity (like acquiring position from a series of velocities), is called *integration*[4].
- Integration and differentiation are inverse[2] operations – if we take a derivative of a quantity, then integrate the derivative, we get back the original quantity. This very important idea is called the *Fundamental theorem of calculus*[5].

1.2.4   Computing a derivative

To show how this works, let's produce some results using the data from Table 1 (page 4). Let's first use position values to acquire a velocity value. To say this another way, let's *differentiate* positions to get velocity:
- Four seconds into the journey the car was located at position 32, and one second later the car was located at position 50. The average velocity between these positions can be acquired by subtracting the second position from the first (or *taking the difference*): 50 - 32 = 18. The difference between two position values is a velocity, a *rate of change* in position.
- Notice that each value in the *average velocity* column represents the difference between the two nearest *position* values.

Expressed simply, *differentiating* a changing quantity means subtracting adjacent values of that quantity to acquire a rate of change (with more refinements explained below).

1.2.5   Computing an integral

Now let's use the data from the table on page 4 to perform the inverse operation – let's *integrate* velocity to acquire position:
- We know the car's position at 4 seconds and our goal is to use average velocities to find its position at 8 seconds.
- To get the desired result, we take the position at 4 seconds and add to it the sum of the average velocities on the interval from 4 to 8 seconds:
  - The initial position at 4 seconds is 32.
  - The sum of the average velocities between 4 and 8 seconds is 18 + 22 + 26 + 30 = 96.
  - Add the initial position to the sum of velocities: 32 + 96 = 128.
  - The result position at 8 seconds is 128.

1.2.6   Relationship between integration and differentiation

Notice about integration that it represents the opposite of differentiation:
- In differentiation we subtract positions from each other to get a *rate of change* in position (a velocity).
- In integration we start with an initial position and add to it a sum of velocities to acquire a new position.

Careful thought will show that differentiation and integration are exact inverses of each other.

1.3   Incremental vs. Continuous Processes

The above example uses a table of stepwise numerical values to roughly approximate a continuous process – a moving car. The table's incremental values are useful in showing a relationship between a process and its rate of change,

but unlike the steps in the table, many natural processes are continuous. In the following sections we learn how to describe a continuous[6] process – a process like those found in nature.

Remember this idea going forward – many natural processes are continuous, therefore our equations should be continuous too.

2 Variables, Equations, Functions

2.1 Variables

In algebra, a letter (or variable[7]) stands for a number – *one letter, any number.* It's a powerful idea, one that allows us to think about symbols that apply to any numbers. In preparation for writing equations able to generate the values listed in Table 1 on page 4, let's choose letters to stand for each of the table's quantities:

- The leftmost table column contains values for time in seconds. Let's use the letter t for this quantity. This is called the *independent variable*[8] because its value doesn't depend on, or result from computations of, the other variables.
- Moving to the right, we have columns for acceleration, velocity and position. Let's use letters a for acceleration, v for velocity, and p for position. These are called *dependent variables*[9] because their value depends on a calculation involving the independent variable t.

Now we can write equations that define, and provide values for, these variables.

2.2 Equations

In the example problem above, we computed results using a table of numerical values, but for most mathematical work an equation is much preferred over a table of numbers. The reasons for this include the fact that an equation is more powerful and general than a number, and because of its universality and simplicity of expression an equation might describe, or even lead to, a physical theory, thus contributing to our understanding of nature.

Having assigned variable names for each of the data table's columns, we can write equations that describe and predict the table's numerical values:

- In the data table on page 4, acceleration is the same for any time argument – this is called a *constant*[10], an unchanging value. Its very simple equation is:

$$a = 4 \tag{2.1}$$

- By examining the table's velocity values, readers may be able to guess the corresponding equation – it appears to be the acceleration constant times time, or:

$$v = at \tag{2.2}$$

- The position equation's form isn't self-evident, but we'll add it to this list while promising to provide a definition below. It is:

$$p = \frac{1}{2}at^2 \tag{2.3}$$

Here are some facts about these equations:

- Just like the numerical quantities described earlier, any of the equations in the above list can be obtained from the others.
- Moving *up* the list from position to acceleration, each equation is said to be the *derivative with respect to time* of the equation below it.
- Moving *down* the list from acceleration to position, each equation is said to be the *integral with respect to time* of the equation above it.
- When expressed in technical language, velocity is the time integral of acceleration, and position is the time integral of velocity[*].
- When expressed in everyday language, the velocity at time t is the *sum of all prior accelerations*, and the position at time t is the *sum of all prior velocities.*
- As before and consistent with the Fundamental Theorem of calculus[5] introduced earlier, if we take the derivative of an equation and then integrate the derivative, we get back what we started with.

The calculus idea from the prior section – that all the numerical quantities can be acquired from any one of them – applies to equations as well: using the methods of calculus, we can acquire all the above equations from any one of them:

- Provided only with the position equation, we can *differentiate* to acquire velocity and acceleration.

---

[*]Position is also described as the "second integral" of acceleration with respect to time.

- Provided only with the acceleration equation, we can *integrate* to acquire velocity and position. But to do this, we need to understand functions.

2.3 Functions

In a powerful idea from algebra, letters stand for numbers – *one letter, any number*. In calculus (and other mathematical fields) there's a similar idea for equations: a *function*[11] stands for an equation – *one function, any equation*. This "symbolic equation" idea is essential for performing calculus operations. Here's an example of functional notation:

$$p(t) = \frac{1}{2}at^2 \tag{2.4}$$

In this example we define $p(t)$, a position function that accepts a time argument and produces a position result. But, just as a variable can stand for any number, *a function can stand for any equation*. This means we can write expressions with functions in much the same way we do with variables.

3 Numerical and Symbolic Differentiation

3.1 Numerical Differentiation

Let's use the above-described function notation to perform numerical differentiation, a calculation based on numerical differences. Specifically, let's perform an operation on the position function shown in 2.4 above to acquire an approximation of its derivative, velocity.

3.1.1 A small step in time

In section 1.2.4, "Computing a derivative" on page 5, we acquired a velocity estimate, a numerical derivative, by subtracting two adjacent positions listed in a data table. Our purpose was to acquire a *rate of change* in position over time. Let's create a more flexible version of that approach by (a) producing results using position function 2.4 defined above, rather than data from a table, and (b) defining a variable representing a *small step in time*. Let's give our time-step this symbol: Δt, pronounced "delta-t". Using this approach and with functional notation we can write:

$$p'(t) \approx \frac{p(t + \Delta t) - p(t)}{\Delta t} \tag{3.1}$$

Some explanation:
- The function $p(t)$, defined in 2.4 above, generates positions for time arguments.
- The expression $p'(t)$ (note the apostrophe) means "the first derivative[3] of $p(t)$", a *rate of change* in position, or velocity[*].
- The symbol "$\approx$" means "approximately equal to". It's present because this numerical method cannot produce exact results.
- The symbol "Δt", pronounced "delta-t", is assigned a small numerical value, sufficient to produce a small step in time and acquire a numerical derivative of position, which is velocity.

To summarize, equation 3.1 uses a method much like that shown in the "Computing a derivative" section above – it computes two position values and subtracts one from the other to get a numerical derivative, an estimate of velocity. The result of the subtraction is then divided by Δt to make the result independent of the chosen step size.

---

[*]The second derivative, acceleration, would be $p''(t)$.
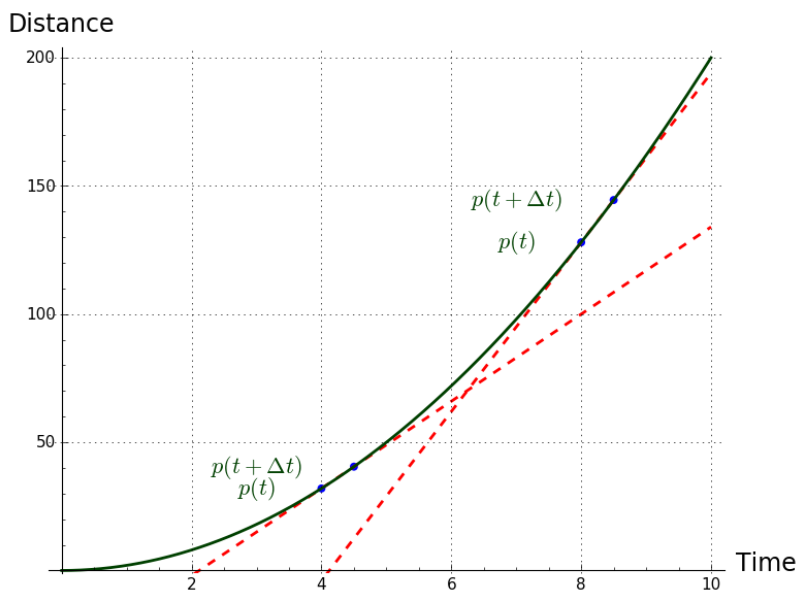
Figure 2: Numerical Differentiation

Figure 2 shows how equation 3.1 looks when plotted. It shows two example rates of change or velocity (dashed red lines) generated by the coordinate pairs equation 3.1 creates, and superimposed on a plot of positions over time. The dashed lines that intersect the position plot are technically known as *tangent lines*[12]. The blue intersection points are intentionally spaced far apart to remind the reader how imprecise numerical differentiation can be.

3.1.2   Approximate Numerical Results

I emphasize again that this method can only produce approximate results. Here's a table of results using function 3.1 with a time argument of 8 and different Δt values:

Δt	Computed Velocity	True Velocity	Error %
0.10000000	32.20000000	32	0.62500000
0.05000000	32.10000000	32	0.31250000
0.02500000	32.05000000	32	0.15625000
0.01250000	32.02500000	32	0.07812500
0.00625000	32.01250000	32	0.03906250
0.00312500	32.00625000	32	0.01953125
0.00156250	32.00312500	32	0.00976563
0.00078125	32.00156250	32	0.00488281
0.00039063	32.00078125	32	0.00244141
0.00019531	32.00039063	32	0.00122070
0.00009766	32.00019531	32	0.00061035

Table 2: Numerical Differential Results

3.1.3   Division by Zero

Table 2 shows that, as we make Δt smaller, the results become more accurate. So why not just set Δt to zero and get perfect results? We can't do that because we must divide by Δt in function 3.1, and in mathematics division by zero[13] is undefined. Here's why – consider this equation:

$$y = \frac{x}{0} \tag{3.2}$$

The result of equation 3.2 is undefined because dividing x by zero (assuming $x \neq 0$) implies the existence of a number y that, when multiplied by zero, would produce x, but there's no such number[*]. Therefore, when using numerical differentiation methods, remember these constraints:

- Δt cannot equal zero, but
- Nonzero Δt values produce errors proportional to their value[†].

These problems are addressed by the idea of a *limit*.

3.2   The Limit

In the above example we assigned a small numerical value to Δt to produce an approximate derivative and discovered that the result's numerical errors decreased as Δt decreased in size (table 2). The idea of examining small intervals is central to calculus, and the limit operator[14] refines this idea – it's a way to make statements about infinitesimally small intervals. Another way to say this is that we can use the limit operator to approach values that cannot be assigned directly, like zero or infinity. For example:

$$\lim_{x \to 0} \frac{1}{x} = \infty \tag{3.3}$$

If the above equation were to be expressed in words, it would be something like, "As x *approaches* zero, $\frac{1}{x}$ *approaches* infinity." This shows how a limit can be used to make statements that aren't constrained by practical or logical limitations such as (in this example) the inconvenient facts that division by zero is undefined and infinity isn't a number[8]. Again, by using a limit, we *approach* specified limits but do not *reach* them.

Among many uses, and in concert with the functional notation introduced above, the limit operator can be used to define a symbolic derivative in terms of a limit:

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{\Delta x} \tag{3.4}$$

In this example, $f'(x)$ (note the apostrophe) is the derivative of $f(x)$ with respect to x, and Δx is a small numerical step (already introduced) that the limit operator makes infinitesimally small.

The limit has a number of applications in both differential and integral calculus, so it merits close study. Applications for this operator are shown in both the Symbolic Differentiation and Symbolic Integration sections.

More importantly, it is by means of the limit operator that we make the transition from stepwise processes to the continuous ones described in section 1.3 above, and write equations that can describe nature.

3.3   Symbolic Differentiation

The most important reason to prefer symbolic differentiation over the numerical methods described above is that the outcome is a symbolic function (and equation) rather than a numerical result. For all the reasons that an equation is more powerful and versatile than a numerical result, the equations and functions produced by symbolic differentiation are much more useful than simple numerical results like those shown above.

Remember that, in both the numerical method described above and in the symbolic method presented here, the expected outcome of the activity is a function that defines a *rate of change* in the original function with respect to a particular variable. Expressed another way, the resulting function will be the symbolic derivative[3] of the original function.

Interestingly, we start the process of symbolic differentiation with almost the same function definition shown in the numerical method (3.1), but with some changes:

$$p'(t) = \lim_{\Delta t \to 0} \frac{p(t + \Delta t) - p(t)}{\Delta t} \tag{3.5}$$

One change is the replacement of "$\approx$" (meaning *approximately equal to*) with "$=$", (meaning *exactly equal to*) because for the majority of functions, a symbolic differentiation is exact. The other change is the presence of the limit operator, which is key to a process of symbolic manipulation that transforms one function into another.

Here are steps in the symbolic differentiation of our example function $p(t)$:

---

[*]Any number, multiplied by zero, equals zero.

[†]Also see the "Computer Processing Limitations" appendix on page 34 for a discussion of the special problems created by computer processing of numerical derivatives.

- For clarity we temporarily abandon functional notation, so the equation within the position function $p(t)$ is revealed:

$$\frac{p(t + \Delta t) - p(t)}{\Delta t} \mapsto \frac{a(\Delta t + t)^2 - at^2}{2\,\Delta t} \tag{3.6}$$

- Now we perform a limit operation, which eliminates Δt and terms multiplied by it and reduces the result to its simplest form (the full process for acquiring this result is shown in the "Detailed Symbolic Differentiation" appendix on page 32):

$$\lim_{\Delta t \to 0} \frac{a(\Delta t + t)^2 - at^2}{2\,\Delta t} = at \tag{3.7}$$

The above process creates a new function that represents velocity and is the derivative of the position function 2.4 defined above. It has this formal definition:

$$p'(t) = \lim_{\Delta t \to 0} \frac{a(\Delta t + t)^2 - at^2}{2\,\Delta t} = at \tag{3.8}$$

The "second" derivative, which defines acceleration, is derived in much the same way:
- Abandon functional notation:

$$\frac{p'(t + \Delta t) - p'(t)}{\Delta t} \mapsto \frac{a(\Delta t + t) - at}{\Delta t} \tag{3.9}$$

- Perform the limit operation and assign the result to a new function identifier:

$$p''(t) = \lim_{\Delta t \to 0} \frac{a(\Delta t + t) - at}{\Delta t} = a \tag{3.10}$$

The symbolic first and second derivatives acquired by this method agree with the definitions established earlier. Again, a more thorough exposition of this method appears in the "Detailed Symbolic Differentiation" appendix on page 32.

4 Numerical and Symbolic Integration

4.1 Numerical Integration

4.1.1 Functions without integrals

Numerical integration plays a larger part in applied mathematics than numerical differentiation does. The reason is that most well-behaved functions have easily acquired symbolic derivatives, but this isn't true for symbolic integrals. Many commonly used functions have no symbolic integrals and must be integrated numerically. Among these are the error function[15] widely used in statistics and other fields –

$$\text{erf}(x) = \frac{1}{2\pi} \int_{-x}^{x} e^{-t^2}\,\mathrm{d}t \tag{4.1}$$

– any orbital problem involving more than two bodies[16], and many others. As a result, much study has been devoted to finding accurate numerical algorithms for these kinds of functions.

4.1.2 Summation of rectangles

The premise of numerical integration is that we can approximate an integral by summing the areas of a series of rectangles. The original function (in this example $f(x)$) provides the rectangle's height, and the summation operator at the right in equation 4.2 provides the summed areas:

$$\int_{a}^{b} f(x)\,\mathrm{d}x \approx \frac{b - a}{m} \sum_{n=1}^{m} f(\frac{n}{m}(b - a) + a) \tag{4.2}$$

Equation 4.2 says that the symbolic integral at the left can be approximated by the summation operation at the right*, using the same function in both cases. This is the same idea we applied earlier in section 1.2.5, "Computing an integral" on page 5 – we took the sum of a series of velocities to acquire a position.

---

*The integration symbol $\int$ refers to a continuous summation of infinitesimal quantities, an "integral," while the summation symbol $\sum$ represents summation of discrete quantities on finite intervals.

4.1.3 Approximate results

It's important to emphasize that numerical integration rarely produces an exact result. The charts in Table 3 show that the outcome accuracy depends on the value chosen for m in equation 4.2, with higher values (i.e. more rectangles) leading to greater accuracy, but – as with numerical differentiation – there are limits imposed by finite computer variable resolution. An additional factor in computer numerical integration is the fact that program runtime increases directly proportional to the m value chosen.
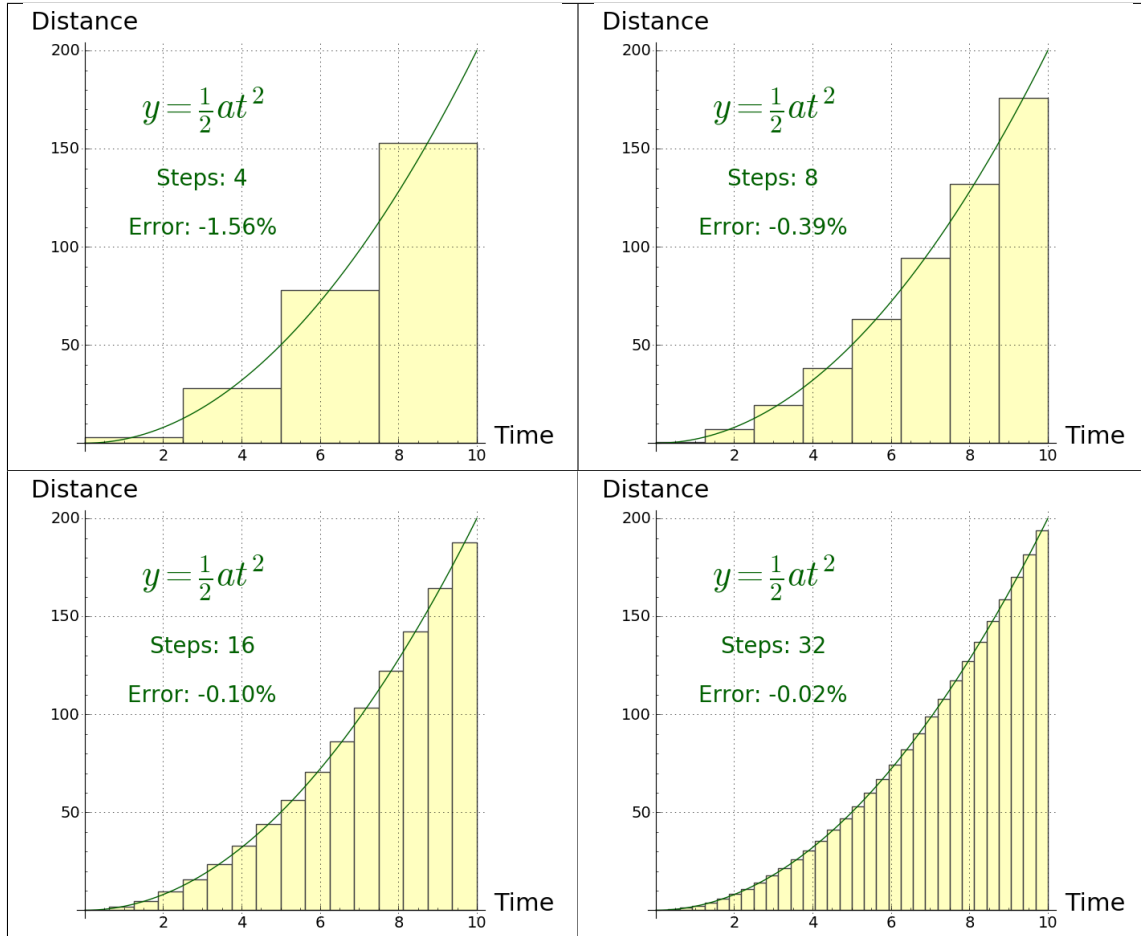
Table 3: Numerical Integration Outcomes

The plots in Table 3 show a green line that represents a continuous process, and a series of rectangles that approximate the process. As explained in section 1.3, the goal is to produce a result that describes a continuous[6] process, rather than approximating it with a series of discrete steps.

4.2 Symbolic Integration

4.2.1 Relation between definite and indefinite integrals

The car example described in section 1.2 on page 4 leads to a solution in terms of *definite integrals*[17], integrals having a defined interval, for example between a and b:

$$\int_a^b f(x)\,dx \tag{4.3}$$

Contrast this with an *indefinite* integral, one having no defined interval:

$$\int f(x)\,dx \tag{4.4}$$

Because the following discussion describes *indefinite* integrals and because our car example relies on *definite* integrals, we show how a definite integral can be obtained from an indefinite one. Let F(x) be the *antiderivative*[19] of f(x), such that:

$$F'(x) = f(x) \tag{4.5}$$

A definite integral is obtained from an indefinite integral like this[17]:

$$\int_a^b f(x) = F(b) - F(a) \tag{4.6}$$

The reader may recall that, in the "Computing an integral" section on page 5, a new position was acquired from a series of velocities by summing (integrating) the velocities, then adding a starting position to the result.

4.2.2 Relation between summation and integration

In contrast to symbolic differentiation, where nearly all functions have symbolic derivatives and are usually easy to process, not all functions have symbolic integrals, indeed many very useful functions fall into the latter category and must be computed numerically.

Equation 4.2 in the numerical integration section shows an approximate relation between summation and integration, but for symbolic integration we can use the limit operator to show the equivalence of a particular kind of summation and an integral[18]:

$$\lim_{\Delta x \to 0} \sum_{x=a}^b f(x)\Delta x = \int_a^b f(x)\,\mathrm{d}x \tag{4.7}$$

This is only meant to symbolize the process in the abstract. Unlike the numerical form shown in equation 4.2, this form is not meant to be applied to real functions as written, and symbolic conversion from functions to their integrals is not so straightforward.

There are several approaches to finding a symbolic integral:

4.2.3 Antiderivative

If function $f(x)$ is the derivative of function $F(x)$:

$$F'(x) = f(x) \tag{4.8}$$

Then $F(x)$ is said to be the *antiderivative*[19] of $f(x)$. This suggests that one way to find an integral for $f(x)$ is to look for a function whose derivative is $f(x)$ – according to the Fundamental Theorem of calculus[5], such a function is the integral of $f(x)$.

Using this method to find integrals can sometimes be difficult and complex, but it has the advantage that a result can be easily verified – we can differentiate a candidate function to see if its derivative is the original function.

4.2.4 Applying Rules

Another approach to finding integrals is to apply established rules. For the car example described in section 1.2 on page 4, we can apply rules to the process of acquiring velocity from acceleration and position from velocity. Earlier we established the identities of the first and second integrals of acceleration by examining a table of numerical values and guessing, but these integrals can also be acquired by the application of rules.

We earlier acquired equations by guessing their identities:
- From acceleration to velocity:

$$\int a\,\mathrm{d}t = at \tag{4.9}$$

- From velocity to position:

$$\int at\,\mathrm{d}t = \frac{1}{2}at^2 \tag{4.10}$$

As it turns out, these pragmatic conversions apply established rules. The first rule is identical to that shown in equation 4.9, while equation 4.10 applies this general exponential integral rule:

$$\int ax^n \mathrm{d}x = \frac{1}{n+1}ax^{n+1}, n \neq -1 \tag{4.11}$$

This rule is applied to equation 4.10 by setting $n = 1$ and noting that $x^1 = x$. The "Rules for Integration" appendix on page 34 lists more such rules and sources for more rules.

4.3   Analyzing a Sphere

This section shows how one can use successive integrations to analyze a three-dimensional object. The analysis adds one dimension per integration – integrating a one-dimensional profile function creates a two-dimensional area, then integrating the area creates a three-dimensional volume.

Apart from their usefulness in teaching calculus, these equations have practical applications. The author uses them to analyze the properties of liquid storage tanks[20] – their surface areas and their partial and full content volumes.

4.3.1   Circle Diameter Function

For this exercise I've chosen a sphere – simple enough to fully understand, but complex enough to be interesting. Let's start with a simple function that generates a bisecting diameter line for a circle with radius r. This simple equation is the basis for all the other forms described in this section:

$$f(t) = 2\sqrt{-(r-t)^2 + r^2} \tag{4.12}$$

For all the functions in this section, the independent variable t has the range $0 <= t <= 2r$. Here's a plot of function 4.12:
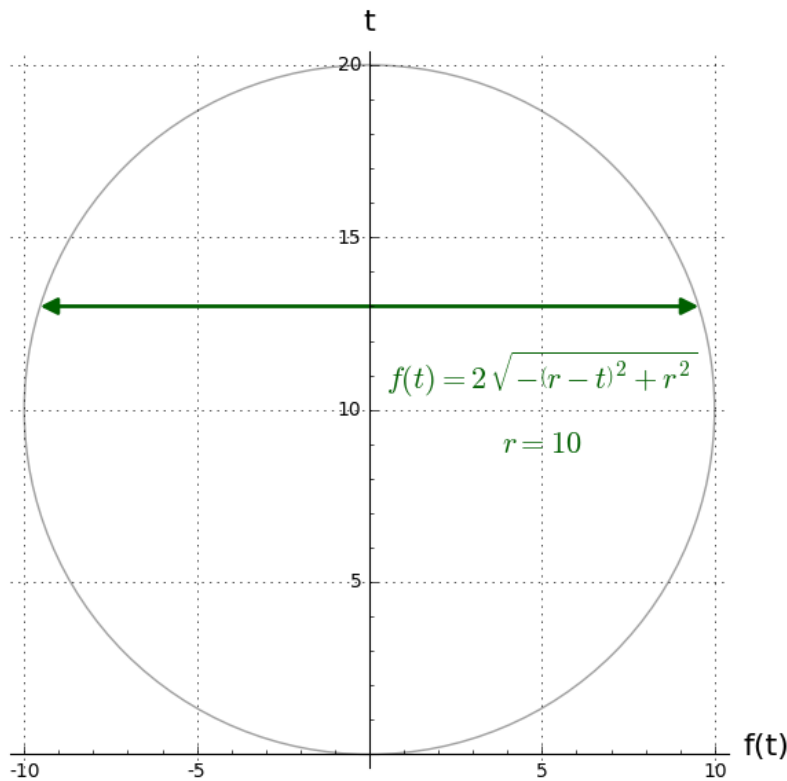
Figure 3: Circle Diameter Function (one dimension)

13

4.3.2 Circle Partial Area Function

To move from one to two dimensions, let's write a function based on 4.12 that, for a given t argument, integrates a series of infinitesimal lines extending across a circle, in the range $0 <= y <= t$. The result should be the partial area of a circle of radius r bisected at t:

$$f(t) = \frac{1}{2}\pi r^2 + \int_0^t 2\sqrt{-(r-y)^2 + r^2}\,dy = \frac{1}{2}\pi r^2 - r^2\arcsin\left(\frac{r-t}{r}\right) - \sqrt{2rt - t^2}(r-t) \tag{4.13}$$

Function 4.13 is often applied to the task of determining the partial volume of a horizontal cylindrical storage tank. A result from 4.13, which in that case would represent the partial surface area of one end of the tank, is multiplied by the tank's length to acquire a partial volume:
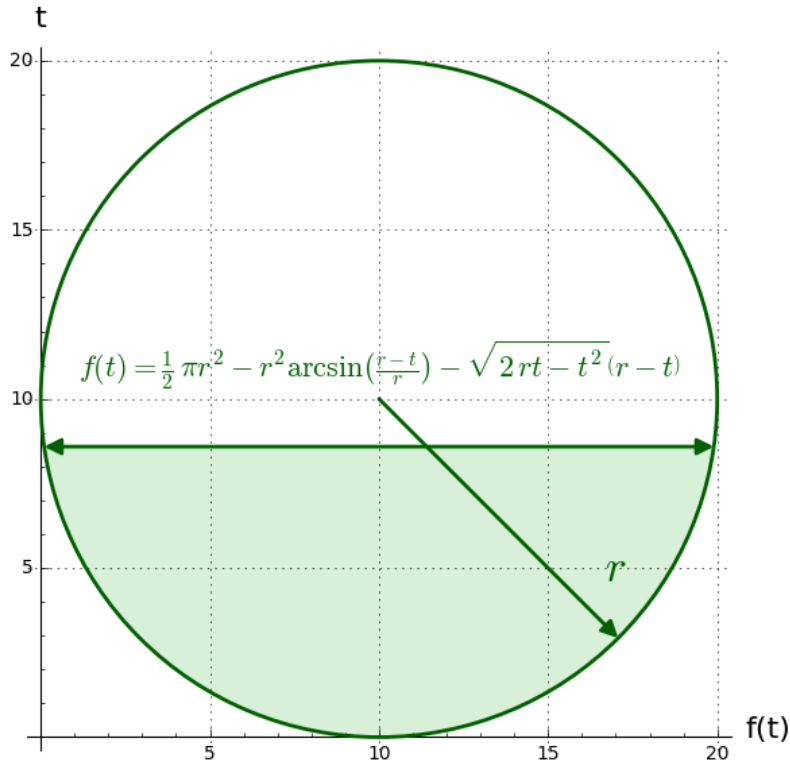
Figure 4: Circle Partial Area Example (two dimensions)

Here's a plot of function 4.13, showing the area profile as a function of t:

$$f(t) = \tfrac{1}{2}\,\pi r^2 - r^2 \arcsin\!\left(\tfrac{r-t}{r}\right) - \sqrt{2\,rt - t^2}\,(r - t)$$

$$r = 10$$
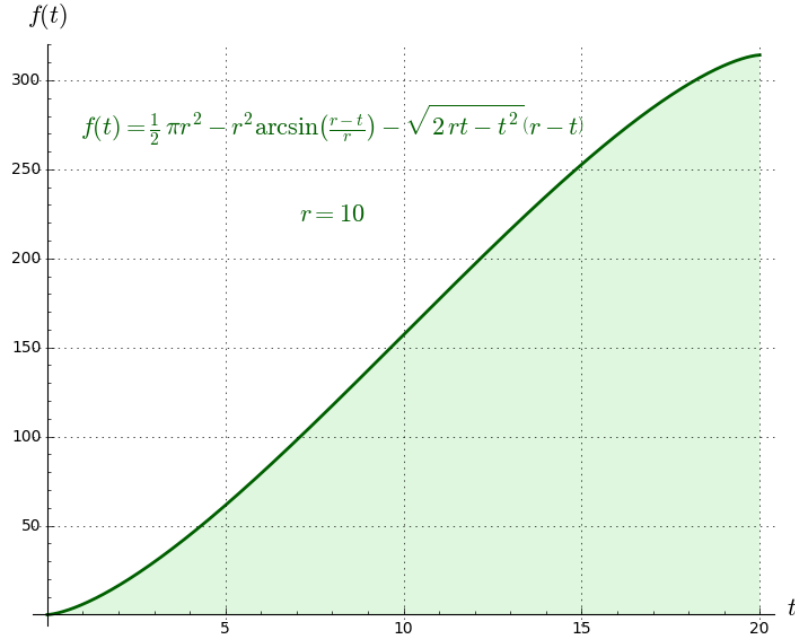
Figure 5: Circle Partial Area Function (two dimensions)

4.3.3 Sphere Partial Surface Area Function

Moving now to three dimensions, the full surface area of a sphere is equal to $4\pi r^2$. Le's add a function, very much like 4.13, to compute partial sphere surface areas based on an argument that bisects the sphere in three dimensions:

$$f(t, r) = 2\pi r^2 + \int_0^t 8\,\sqrt{-(r - y)^2 + r^2}\,\mathrm{d}y = 2\,\pi r^2 - 4\,r^2 \arcsin\left(\frac{r - t}{r}\right) - 4\,\sqrt{2\,rt - t^2}(r - t) \tag{4.14}$$

4.3.4 Sphere Partial Volume Function

Finally, a surprisingly simple function that computes partial sphere volume. A full sphere volume should be $\tfrac{4}{3}\pi r^3$, let's see how this turns out:

$$f(t) = \pi \int_0^t -(r - y)^2 + r^2\,\mathrm{d}y = \pi r t^2 - \frac{1}{3}\,\pi t^3 \tag{4.15}$$

Here's a sphere volume profile for function 4.15:

15
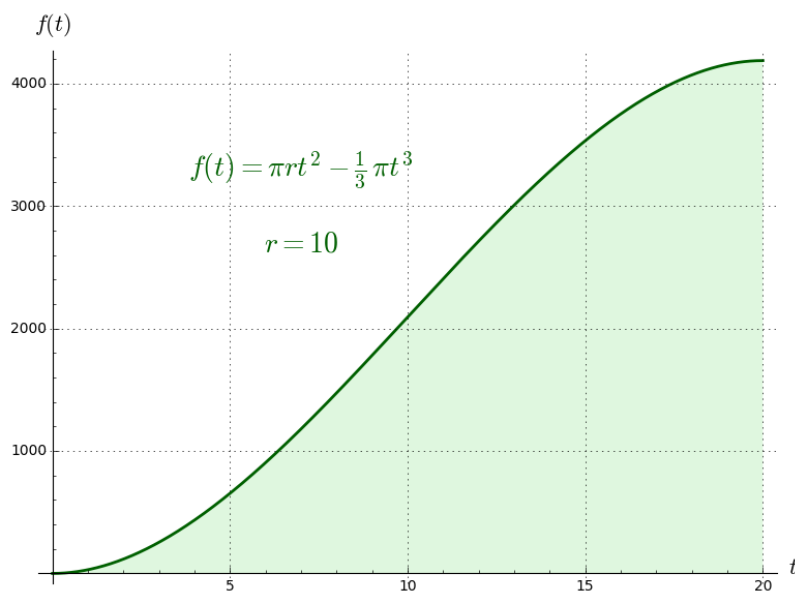
Figure 6: Sphere Partial Volume Function (three dimensions)

Given a basis like the partial circumference equation shown in 4.12 and with the assistance of Computer Algebra Systems (page 19) to acquire efficient integrals, it's not difficult to assemble a full analysis of a given three-dimensional shape.

5   Differential Equations

A differential equation[21] is one that relates a function to its derivatives. Differential equations are very important to modern science and technology. It's only a small exaggeration to say that modern physical theory is assembled from a large set of differential equations.

5.1   First-Order

5.1.1   Rate of Decline

Let's start with a very common first-order[22] differential equation (hereafter DE) that models natural growth and decay. It looks like this:

$$y'(t) + y(t)\,r = 0 \tag{5.1}$$

$$y(0) = C \tag{5.2}$$

Where:
- t = time, the independent variable.
- $y(t)$ = an unknown function of t.
- $y'(t)$ = the first derivative of y with respect to time[*].
- r = a value that determines the function's rate of change with respect to time.
- C = a constant that establishes an *initial value*[23] at time zero.

The solution to this DE is:

$$y(t) = Ce^{-rt} \tag{5.3}$$

Where:
- e = the base of natural logarithms[24].
- C = a constant of integration whose value depends on specific circumstances.

---

[*]There are many derivative notations, some described in the "Calculus Notations" appendix on page 35.

When plotted as a function of time, the result looks like this:

Amplitude

$$y(t) = Ce^{-rt}$$

$$C = 10, r = 1.00$$
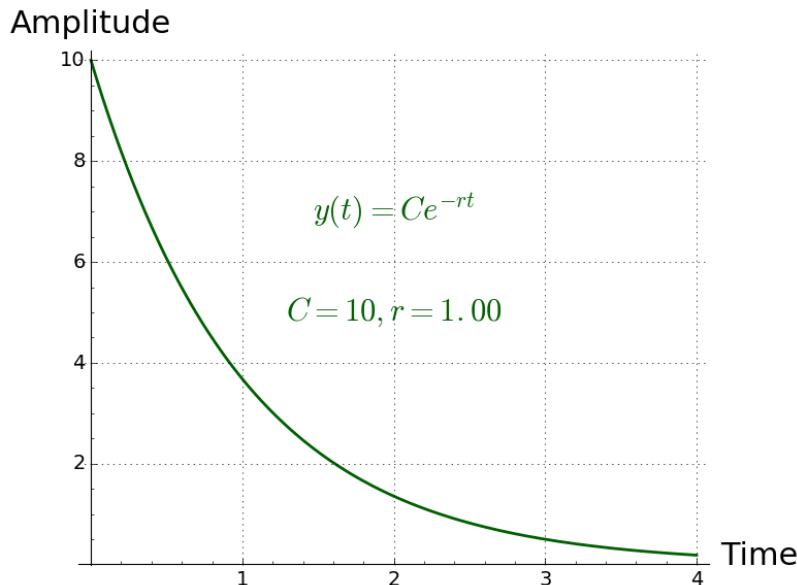
Figure 7: Plot of Natural Decay Equation

Here are some notes on this example:

- A DE consists of one or more statements that describe an unknown function.
- Solving a DE is much like integrating a function, in fact symbolic or numerical integration is usually involved in the solution.[*]
- Not all DEs can be solved symbolically, but as with integrals, numerical methods can be used to acquire approximate results.
- This DE is called "first-order" because it only includes a first derivative. See below for a second-order DE.
- This specific DE is widely applied in fields such as physics, biology, electrical engineering and thermodynamics among others, to describe natural processes of decline – the rate of change in temperature between two connected reservoirs of heat energy, the rate of water flow between two connected bodies of water, the rate of gas pressure change between two connected, pressurized gas reservoirs, the rate of voltage change in an electrical resistor-capacitor circuit, the rate of decline in sources of radioactivity, and many others.
- Intuition into the behavior of this DE can be had by noting that the function's first derivative or rate of change – $y'(t)$ – is determined by a constant value assigned to r, therefore as the value of $y(t)$ declines, the *amount* of change declines in proportion. This means that in theory, for a nonzero C initial value, the value of $y(t)$ never becomes zero.

5.1.2 Rate of Growth

In this example, we make a very small change in the unknown function's definition to track rates of growth:

$$y'(t) - y(t)\, r = 0 \tag{5.4}$$

$$y(0) = C \tag{5.5}$$

Notice that the only change from definition 5.1 above is from $y'(t) + y(t)r$ to $y'(t) - y(t)r$, with significant changes in the behavior of the solution.

Here is the solution to this DE:

$$y(t) = Ce^{rt} \tag{5.6}$$

---

[*]Because this article is an introduction to calculus, it doesn't cover the specifics of solving DEs, except by way of computer algebra systems described below.

Note that the only change from solution 5.3 above is that a negative exponent becomes a positive one. Here is a plot of the solution as a function of time:
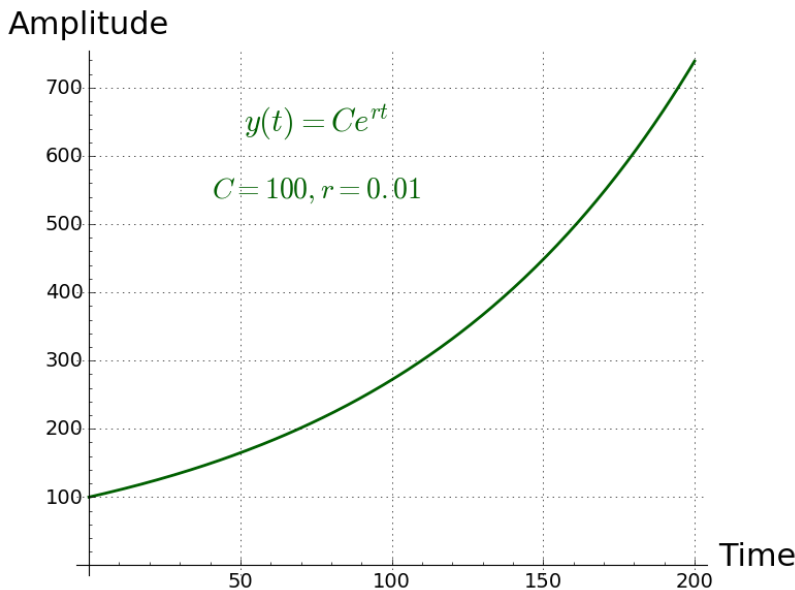
Figure 8: Plot of Natural Growth Equation

Here are some notes on this DE:

- This DE differs from the first in that the rate of change is positive, so the amount of change $y'(t)$ over time *increases* and is proportional to the value of $y(t)\,r$. As before, the rate of change with respect to time is determined by a constant value assigned to r.
- This DE has the property that, for a value $y(t)$ at any time t, that value will have doubled after $\frac{\log(2)}{r}$ units of time[*].
- This DE is widely applied to problems involving biological growth, population studies[25], compound interest, and other systems that involve patterns of natural growth.
- Figure 8 shows that, for a growth rate r of 1% per unit of time (i.e. $r = 0.01$), the function's value doubles in $\frac{\log(2)}{r}$ or 69.3 units of time, and quadruples in $\frac{\log(4)}{r}$ or 138.6 units of time.

5.2   Second-Order

5.2.1   Harmonic Oscillator

Second-order DEs, DEs that involve second derivatives, are somewhat more complex than first-order, so we'll show just one relatively simple and well-studied example. Here is its definition:

$$y''(t) + y(t) = 0 \tag{5.7}$$

$$y(0) = 0 \tag{5.8}$$

$$y'(0) = 0 \tag{5.9}$$

$$y''(0) = 1 \tag{5.10}$$

Before we solve this DE, I ask my readers to think about the system described in equations 5.7 through 5.10:

- As 5.7 is written, it seems that $y''(t)$ will always be the negation of $y(t)$. Think about what $y''(t) + y(t) = 0$ means – it means $y''(t)$ will always be equal to $-y(t)$ for nonzero values. For example, when $y(t) = 1$, $y''(t) = -1$ and vice versa.

---

[*]We can say that, for a given rate of increase r, the value will have increased by N times in $\frac{\log(N)}{r}$ units of time.

- In summary, this DE defines a closed loop, with $y(t)$ getting its rate of change from $y'(t)$ and $y'(t)$ getting its rate of change from ... wait for it ... $-y(t)$.
- On that basis, we can predict that this DE will oscillate endlessly, and that is exactly what happens.

Here's the solution to the DE defined by 5.7 - 5.10:

$$y(t) = sin(t) \tag{5.11}$$

Because $y'(t)$ is not specified in the definition and because it plays a part in the result, here is the first derivative of $y(t)$:

$$y'(t) = cos(t) \tag{5.12}$$

Here is a plot of this second-order DE, including traces for both $y(t)$ and $y'(t)$:
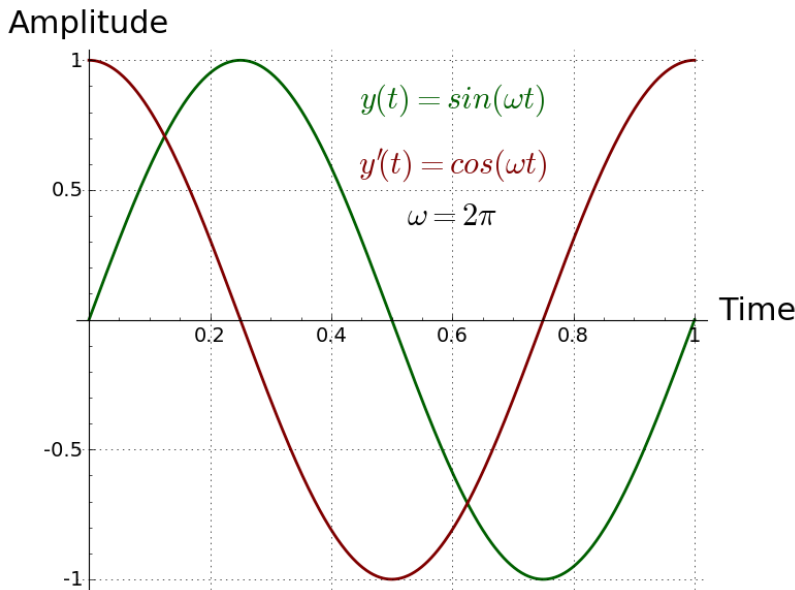
Figure 9: Plot of Second-order DE (harmonic oscillator)

With a bit of insight, the reader can see that the brown first derivative trace ($y'(t)$) represents a rate of change in the primary trace $y(t)$. Not plotted but implied by equation 5.12 is that $y''(t)$ equals $-sin(t)$.

6 Computer Algebra Systems

6.1 Rationale

Many of the procedures described in this article can be automated or greatly simplified by use of a special kind of computer software called a *Computer Algebra System* (CAS)[26]. A modern CAS can provide a variety of numerical and symbolic results that were not practical until recently. And better, such programs are becoming less expensive and in many cases are free.

The objection is sometimes made that a CAS shields its users from the underlying mathematics and may produce results while providing no insight into the reasoning behind them, but there's a counterargument – that by its ease of use, accessibility and visual aids a CAS may teach mathematics that users might not otherwise understand. To support this point, let me ask how many people would learn calculus if they were obliged to perform long division with a pencil and a pad of paper?

6.2 CAS Examples

The example environments in this section are limited to those able to perform symbolic calculus, and because many of my readers will be students I favor the least expensive solutions. For the sake of comparison I include a Mathematica example, even though because of its high cost I don't recommend it to people learning mathematics.

6.2.1   Python/SymPy

A Python[27] library named "SymPy"[28] is able to provide symbolic solutions to many kinds of mathematical problems and is a lightweight solution – only a small amount of computer power is required[*].

Here's a symbolic solution to the Harmonic Oscillator second-order differential equation example (page 18), performed using SymPy's symbolic functions, with this example carried out in the (optional) Jupyter[29] interactive notebook environment:

```
from sympy import *
var('a:z C1 C2');
y = Function('y')
de = diff(y(t),t,t) + y(t)
f = Lambda((t,C1,C2)  , dsolve(de,y(t)).args[1])
p = plot(f(t,1,0),f(t,0,1),(t,0,4*pi),
   show=False, title='$'+latex(de)+'=0$')
p[0].line_color = 'DarkGreen'
p[1].line_color = 'DarkRed'
p.show()
```

Figure 10: Python/SymPy second-order DE listing (harmonic oscillator)
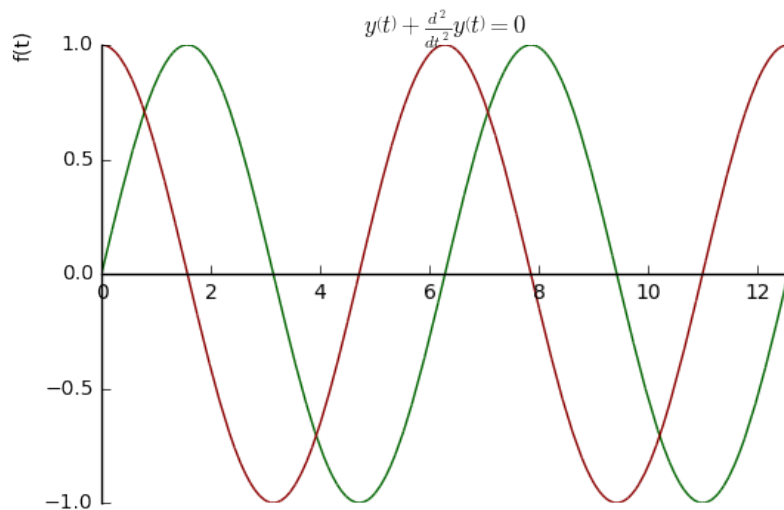
Figure 11: Python/SymPy second-order DE plot (harmonic oscillator)

Some notes about Figure 11:
- Notice the second-order derivative notation in the title of the plot: $y(t) + \frac{d^2}{dt^2} y(t) = 0$. It seems that SymPy uses a different notation for derivatives than that used in this article. I emphasize again that all such notations are equivalent[10.5] and a matter of taste.
- This example was carried out in Jupyter[29], a rather nice open-source interactive notebook environment, an environment that greatly speeds up mathematical explorations and development. But the same code placed in a plain-text Python source file produces the same result, so a lightweight computing environment can produce this result as well.
- The above result can be acquired using a command-shell based embodiment of SymPy called ISymPy[30] which reduces resource requirements to an absolute minimum.
- At the time of writing there's an online interactive version of SymPy[31] in a web page, a truly minimal-resource solution.
- Each CAS has a different notation for expressing calculus ideas – there's a learning curve for each of them. Some environments can solve more equations than others.

---

[*]A Raspberry Pi can easily handle the processing required by Python/SymPy, but not necessarily the Jupyter notebook environment shown in this example.

- Of the CAS examples in this section, this Python/Sympy CAS example requires the least computer power (and even less if the command-line option is chosen).

The Jupyter notebook environment is a powerful tool that speeds up mathematical and technical work, and it supports any number of different "kernels" or language processors[32], even simultaneously. This example could have been created without Jupyter, but it would have taken more time.

6.2.2 SageMath

The SageMath project[33], formerly known as Sage, is one I've written about in the past[34]. SageMath is an ambitious project and has a much larger storage requirement than Python/Sympy, typically 4-5 gigabytes installed.

SageMath is a bundle of different mathematical resources* configured to be accessible through an interface based on Python. Unlike the Python/Sympy environment, SageMath performs more preprocessing of entered expressions, so mathematical expressions are easier to enter and manipulate.

In a recent development SageMath has been made compatible with the Jupyter notebook environment, and this example, like the first, produces a result using Jupyter.

```
var(map(chr,xrange(ord('a'),ord('z'))));
y = function('y')
de = diff(y(t),t,t)+y(t)==0
f(t,_K1,_K2) = desolve(de,y(t),t)
p = plot(f(t,0,1),(t,0,4*pi),figsize=5,color='#800000'
     ,gridlines=True,title='$'+latex(de)+'$')
p += plot(f(t,1,0),(t,0,4*pi),color='#006000')
p
```

Figure 12: SageMath second-order DE listing (harmonic oscillator)
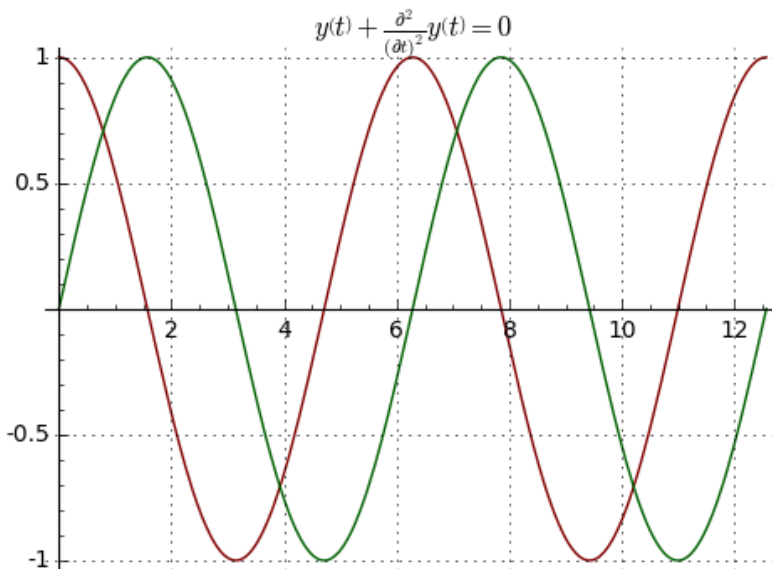
Figure 13: SageMath second-order DE plot (harmonic oscillator)

I want to emphasize about Figure 13 that it very much sells SageMath short. I wanted to show the same symbolic processing example for all the environments, but this makes SageMath look much less capable than it really is. The truth is that, in typical mathematical work, SageMath is much easier to use than Python/Sympy and other similar environments like Maxima[35] (on which SageMath relies internally for its symbolic results). SageMath has been in active development for about 12 years at the time of writing and includes many packages meant to satisfy the needs of those involved in high-level mathematical research and education. It also can solve more kinds of equations than Python/SymPy can.

---

*Including NumPy, SciPy, matplotlib, Sympy, Maxima, GAP, FLINT, R "and many more" according to the SageMath website[33].

6.2.3 Mathematica

I include this entry out of a desire for completeness, even though I don't recommend Mathematica[36] for most students or mathematicians because of its great cost (at the time of writing, a one-seat non-student license is approximately US$2500.00). Mathematica can solve a subset of problems out of reach of most other CAS solutions, but for most people the cost makes it unacceptable.

6.2.4 Skydiver equation

For balance I'll show a problem that only Mathematica can solve, not Sympy nor SageMath – the "Skydiver equation". It's a second-order DE that models the dynamic behavior of a body falling through the atmosphere. For this problem the DE must model the fact that a high initial acceleration declines as air resistance builds up at higher velocities, until gravitational acceleration is exactly balanced by air resistance and velocity becomes a constant, a "terminal velocity". Here's a statement of the DE:

$$y''(t) = y'(t)^2 k - g \tag{6.1}$$

$$y'(0) = 0 \tag{6.2}$$

$$y(0) = a \tag{6.3}$$

Where:
- t = time, seconds.
- $y(t)$ = position, altitude in this system, at time t.
- g = Earth's gravitational acceleration, approximately 9.8 $m/s/s$, known as ("little-g"[37]).
- a = initial altitude, meters.
- k = a constant that takes into account the external shape and properties and ratio of mass to surface area of a falling object.

Equation 6.1 describes an acceleration profile that balances g, Earth's gravitational acceleration, against air resistance, which increases as the square of velocity, the latter represented by $y'(t)^2 k$.

Equation 6.2 says that the initial velocity (at time zero) is zero.

Equation 6.3 says that the initial position is a, a provided altitude, presumably the height of the airplane.

This is the most complex DE in this article, and to date among a handful of the most complex equations I've tried to submit to CAS methods for solution. And as already pointed out, SymPy and SageMath cannot solve it. Here's a narrative of Mathematica's solution:

- The statement of the DE as submitted to Mathematica:

$$\text{de} = \left\{ y''(t) = ky'(t)^2 - g, y(0) = a, y'(0) = 0 \right\} \tag{6.4}$$

- The Mathematica expression to solve it[*]:

$$\text{Quiet}[\text{sol} = \text{DSolveValue}[\text{de}, y(t), t]] \tag{6.5}$$

- Mathematica's solution for position:

$$y(t) = \frac{ak - \log\left(\cosh\left(\sqrt{g}\sqrt{k}\,t\right)\right)}{k} \tag{6.6}$$

- Velocity:

$$y'(t) = -\frac{\sqrt{g}\tanh\left(\sqrt{g}\sqrt{k}\,t\right)}{\sqrt{k}} \tag{6.7}$$

- Acceleration:

$$y''(t) = -g\,\text{sech}^2\left(\sqrt{g}\sqrt{k}\,t\right) \tag{6.8}$$

---

[*]The "Quiet" function suppresses a long list of cautionary messages Mathematica produces about its solution

- "Jerk"[*] (rate of change in acceleration):

$$y'''(t) = 2g^{3/2}\sqrt{k}\tanh\left(\sqrt{g}\sqrt{k}\,t\right)\operatorname{sech}^2\left(\sqrt{g}\sqrt{k}\,t\right) \tag{6.9}$$

Here are plots of the solutions for each derivative:

(a) Skydiver Position $y(t)$

(b) Skydiver Velocity $y'(t)$

(c) Skydiver Acceleration $y''(t)$

(d) Skydiver "Jerk"[38] $y'''(t)$
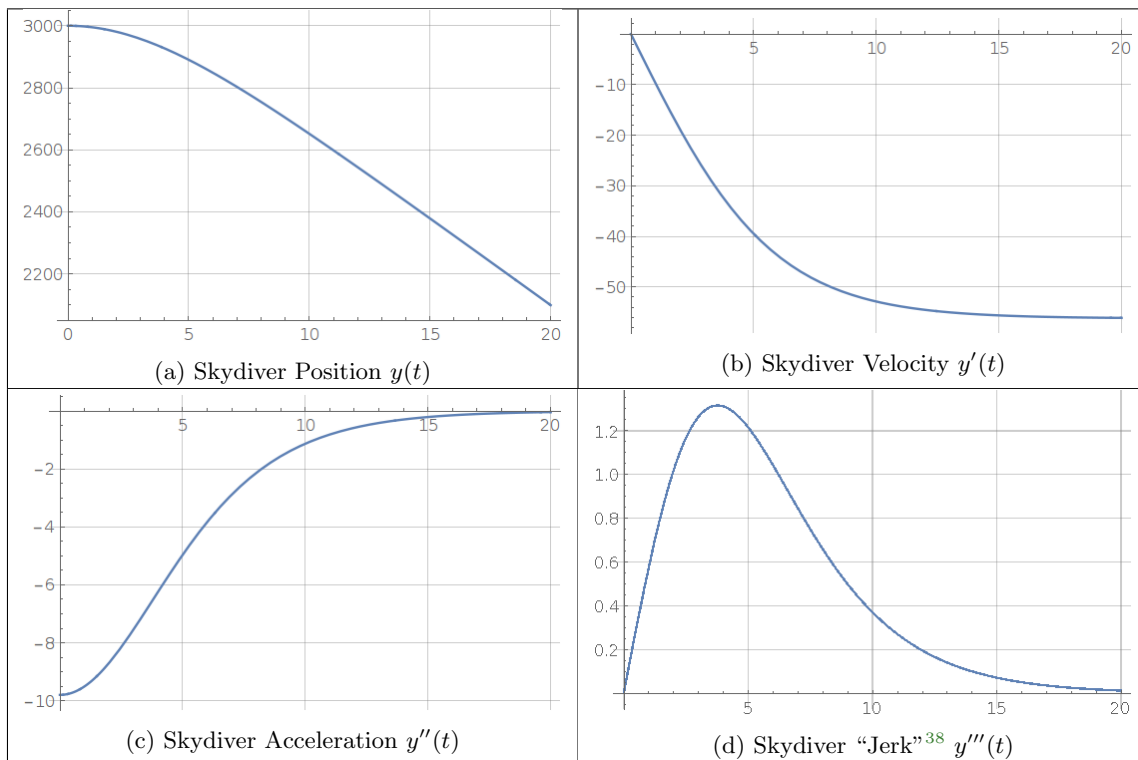
Figure 14: Skydiver Result Plots

The plots in Figure 14 show that, after falling for about 15 seconds, a skydiver has nearly reached terminal velocity. These plots provide insight into the relationship between a function and its derivatives – for a function $y(t)$ whose value changes at a steady rate (like position after about 15 seconds), its first derivative $y'(t)$ should be a constant, and its second derivative $y''(t)$ should be zero.

As highly regarded as Mathematica is and despite much effort, I can't persuade it of something as simple as the idea that $\sqrt{g}\sqrt{k} = \sqrt{gk}$ for positive, non-complex g and k values. For US$2500.00 one would think something as trivial as that should be achievable.[†]

Just to further complicate things, a free copy of Mathematica is included with Raspbian[39], a Linux distribution designed for the small, inexpensive but powerful Raspberry Pi[40] computers. It seems that Wolfram Research[41] (the source of Mathematica) would like to entice younger, more impressionable minds into eventually buying a more expensive version of their product.

6.3 Conclusion

For those seeking a CAS solution and after re-evaluating all the choices available at the time of writing, my advice for the majority of readers is to install Jupyter and SymPy. An alternative choice, for a few with particular needs, is to install Jupyter with Sage and SymPy (a much larger installation). Here are installation notes for each option:

6.3.1 Jupyter/SymPy

For Windows. Linux and the Macintosh, readers may acquire an installer from the Anaconda download page located at `https://www.continuum.io/downloads`[42]. The installation procedure is much the same on all the platforms. At

---

[*]Honestly, that's what it's called[38].

[†]In fairness, neither SymPy nor SageMath can be persuaded of this either.

the time of writing Anaconda installs Python 3.5 with Sympy, SciPy, NumPy and a number of other useful libraries, plus the Jupyter notebook interface shown in examples above. Linux users can install Jupyter/SymPy without relying on Anaconda, but if the system has adequate storage space, there's no need to fine-tune the process.

6.3.2 Jupyter/SymPy/Sage

Because Sage is so much larger than the other CAS packages, and because of its well-known connection with the Linux platform, for those running Windows it's simpler to acquire and install it as a virtual machine[43]. As it happens the people behind SageMath have anticipated this and offer something they call the SageAppliance[44], which along with VirtualBox[45] allows Windows users to run SageMath locally, by creating a Linux virtual machine to host SageMath.

Those running Linux have more options – they can download an installable SageMath binary[33] for most Linux distributions. And on Linux, Jupyter is easier to install as well – there are several approaches apart from the Anaconda installation method described above.

7 Numerical Methods

To re-emphasize a point made earlier, there are many equations and functions that cannot be expressed analytically and that must be solved numerically. Even though there's a long list of such functions, for this example we'll look at a problem that can be solved analytically, in order to validate a numerical result by comparing it to the analytical one.

7.1 Numerical Harmonic Oscillator

The operation of a well-designed numerical algorithm should be as self-documenting as practical, and this example meets that requirement. Here's a result from a Jupyter[29] notebook session (using the Python kernel) showing a numerical solution to the harmonic oscillator DE presented earlier in analytical form:

```
from matplotlib import pyplot as plt
import numpy as np
π = np.pi
plt.clf()
y = 0
vy = 1
m = 1000
Δt = 1.0 / m
f = 2
ω = 2 * π * f
ta = []
ya = []
yva = []
for t in range(m):
    vy += y * ω * Δt
    y -= vy * ω * Δt
    ta += [t/m]
    ya += [y]
    yva += [vy]
plt.plot(ta,yva,color='DarkRed')
plt.plot(ta,ya,color='DarkGreen')
plt.xlabel('Time␣seconds')
plt.ylabel('Amplitude')
plt.title('$y(t) + \\frac{d^2}{dt^2} y(t) = 0$')
plt.grid()
plt.show()
```

Figure 15: SageMath second-order DE listing (harmonic oscillator)

24
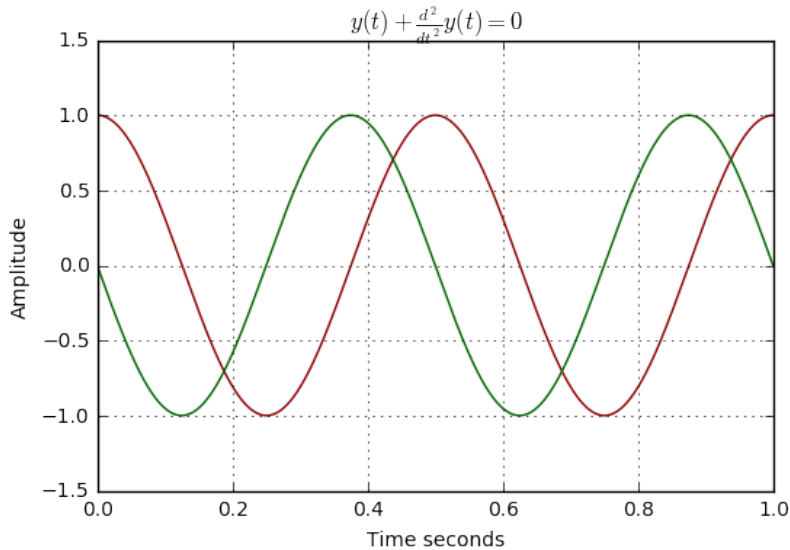
Figure 16: Numerical second-order DE solution (harmonic oscillator)

7.1.1 Explanation

Here's the essence of the algorithm shown in Figure 16:

     1. `vy += y * ω * Δt`
     2. `y -= vy * ω * Δt`

In line (1), velocity variable `vy` is updated by position variable `y` times Δt, a small value whose purpose was explained earlier, and ω, a value that sets the frequency of oscillation. In line (2), position variable `y` has `vy` subtracted from it, times Δt and ω. Just as with the analytical harmonic oscillator DE $y(t) + \frac{d^2}{dt^2}y(t) = 0$, this produces an oscillating system.

7.1.2 Parallel with Numerical Integration/Differentiation

There's a strong parallel between a numerical equation solver like this, and numerical integration and differentiation, indeed many of the same methods are used. And just as with those methods, a smaller value for Δt produce more accuracy at the cost of longer running times, and with the caveat that a very small value will begin to produce numerical errors, for reasons given in the Computer Processing Limitations appendix on page 34.

7.2 Numerical Elliptical Orbit

This section discusses a numerical algorithm that, by modeling a planetary orbit, can be used to test a physical theory – energy conservation[46]. According to this theory, energy is neither created nor destroyed, only changed in form. If this theory is true, it should be verifiable by examining elliptical planetary orbits.

    A planetary orbit has two kinds of energy – kinetic energy[47] (KE), the energy of motion, and gravitational potential energy[48] (GPE), an energy proportional to position.

7.2.1 Kinetic Energy

Kinetic energy has this definition:

$$KE = \frac{1}{2}mv^2 \tag{7.1}$$

    Where:
- KE = Kinetic energy, Joules.
- m = mass, kilograms.
- v = velocity, meters/second.

7.2.2 Gravitational Potential Energy

Gravitational potential energy has this definition:

$$GPE = -G\frac{m_1 m_2}{r} \tag{7.2}$$

Where:
- GPE = Gravitational potential energy, Joules.
- G = Universal gravitational constant[49].
- m_1, m_2 = Masses of orbiting bodies, kilograms.
- r = The distance between the orbiting bodies, meters.

7.2.3 Negative Value for GPE

GPE values calculated in physics, and in in equation 7.2, are by convention either negative or zero – here's why:
1. At an infinite separation between bodies, GPE should equal zero.
2. Lifting a mass in a gravitational field, or increasing the radius r between m_1 and m_2 in equation 7.2, requires an expenditure of energy, so with increasing separation the GPE value should become more positive, or less negative (which amounts to the same thing).
3. To simultaneously satisfy points (1) and (2) above, GPE must be negative everywhere but at infinity, where it should be zero.

7.2.4 Constant Energy

This orbital problem is sufficiently complex that numerical methods must be used to model it. For an elliptical orbit, as time passes the distance between the orbiting bodies changes, as do their velocities, so both KE and GPE are changing. If conservation of energy is a valid theory and barring friction and other energy losses, these two changing energies should sum to a constant at all points in the orbit, i.e. $KE + GPE = C$. That's the first theory we'll be testing in this model – is orbital energy a constant?

7.2.5 Kepler's Second Law

This simulation tests another theory – Kepler's Second Law[50]: "A line segment joining a planet and the Sun sweeps out equal areas during equal intervals of time." It turns out that, as written, this model can test both theories at once.

7.2.6 Program Listing

The model was written in Python, using the Jupyter notebook environment to speed up the process of testing and revision. Here's the program listing:

```python
from matplotlib import pyplot as plt
G = 6.673e-11 # universal gravitational constant
sm = 2e30 # solar mass kilograms
m = int(2e6) # steps in simulation
md = m/16
Δt = 1.2e7/m # time step

def polygon_area(array):
    a = 0
    ox,oy=0,0
    for x,y in array:
        a += y * ox - x * oy # determinant
        ox,oy = x,y
    return a/2.0

plt.clf()
plt.figure(figsize=(8,4))
op = complex(149e9,0) # initial orbital position
ov = complex(0,1e4) # initial orbital velocity
gpr = []; gpi = []
poly = [[0,0]]
polgx = [0]; polgy = [0]
state = 0
print("   %-16s %-16s %-16s %-16s" % ('KE', 'GPE', 'KE+GPE', 'Area'))
print("-" * 70)
for n in range(m):
    poly += [[op.real,op.imag]]
    if(n % 1000 == 0):
        polgx += [op.real]; polgy += [op.imag]
        gpr += [op.real]; gpi += [op.imag]
    # compute gravitational acceleration
    a = -G * sm * op * (abs(op)**2)**-1.5
    ov += a * Δt # increment velocity
    op += ov * Δt # increment position
    if(n % md == 0):
        state += 1
        if(state % 2 == 0):
            poly += [[0,0]]
            area = polygon_area(poly)
            polgx += [0]; polgy += [0]
            plt.plot(polgx,polgy,color='gray')
            plt.fill(polgx,polgy,alpha=0.2)
            # plot polygon label
            tx = (polgx[1] + polgx[-2])/2.5
            ty = (polgy[1] + polgy[-2])/2.5
            lbl = chr(ord('@') + int(state/2))
            plt.text(tx,ty,lbl)
            opt =  op - ov * Δt * 0.5 # intermediate position
            ke = (abs(ov)**2.0)/2.0 # compute kinetic energy
            pe = -G*sm/abs(opt) # compute potential energy
            print("%c %16.9e %16.9e %16.9e %16.9e"
                  % (lbl,ke,pe,ke+pe,area))
        else:
            poly = [[0,0]]
            polgx = [0]; polgy = [0]

plt.plot(gpr,gpi,color='gray') # plot orbit
plt.plot(0,0, 'oy') # plot solar position
plt.axis('equal') # assure equal figure proportions
plt.title('Elliptical Orbit Analysis')
plt.grid()
plt.show()
```

Figure 17: Elliptical Orbit Numerical Model Program Listing

7.2.7 Results

Here is the program's results table:

	KE	GPE	KE+GPE	Area
A	5.915025527e+07	−9.048549533e+08	−8.457046980e+08	5.587455300e+20
B	1.423514785e+08	−9.880561765e+08	−8.457046980e+08	5.587455300e+20
C	3.919651862e+08	−1.237669884e+09	−8.457046980e+08	5.587455300e+20
D	1.510712220e+09	−2.356416918e+09	−8.457046980e+08	5.587455300e+20
E	1.628050668e+09	−2.473755366e+09	−8.457046980e+08	5.587455300e+20
F	4.076256220e+08	−1.253330320e+09	−8.457046980e+08	5.587455300e+20
G	1.475607164e+08	−9.932654144e+08	−8.457046980e+08	5.587455300e+20
H	6.055581909e+07	−9.062605171e+08	−8.457046980e+08	5.587455300e+20

Figure 18: Elliptical Orbit Model Results

Note in Figure 18 that the KE+GPE result columns, showing the sum of kinetic and potential energy, agree with each other to ten decimal places, which confirms the theory of conservation of energy, and the area column values also agree, which confirms Kepler's Second Law.

This example is meant to show how mathematics can reveal connections between theories and objectively quantify predictions of theories, with an efficiency unequaled by other kinds of reasoning. We used mathematics to test a corollary of the theory of energy conservation and our result is perfectly clear: orbits conserve energy.

7.2.8 Orbital Plot

Here's the orbital plot the program generates:
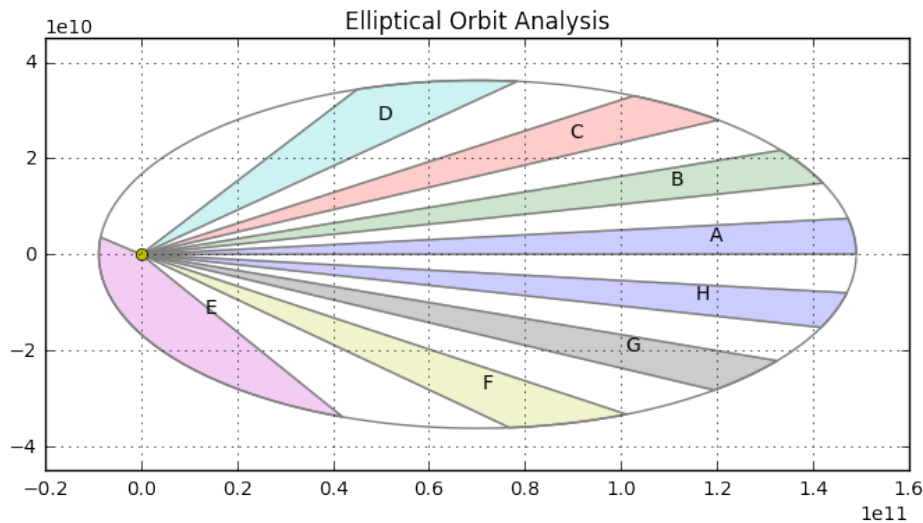
Figure 19: Elliptical Orbit Analysis Diagram

In Figure 19 the system's central body is shown as a yellow circle at the left. Also shown are the polygons included to test Kepler's Second Law[50] – in spite of their varied shapes, all their areas are equal.

8 Infinity

The idea of infinity appears frequently in various mathematical contexts, so it deserves an explanation. First, infinity is not a number – one cannot multiply or divide a number by infinity and acquire a meaningful result. To see why this is so, consider the Infinity Hotel.

8.1 The Infinity Hotel

Welcome to the Infinity Hotel – we always have room for more guests, no one is turned away. If you show up without a reservation on a busy Saturday night, no problem – we'll just put you in room one. But the Infinity Hotel always

appears to be full, so how can we put you in room one? That's easy – we first move the occupant of room one to room two. But to do that, we move the occupant in room two to room three, *ad infinitum.*

After you've occupied room one, after the other guests have been moved to make room for you, the Infinity Hotel is the same size it was before – it's still infinitely large.

This is why we can't multiply a number by infinity and expect a meaningful result. Just as with the idea of dividing by zero, to multiply a number n by infinity implies the existence of a number that, if divided by infinity, would produce n, but no such number exists.

8.2 Infinite Summation

One may ask, if infinity isn't a number and can't appear in an equation in a numerical context, then what purpose does it serve in mathematics? Easily explained – consider this equation[*]:

$$y = \sum_{n=1}^{m} 2^{-n} \tag{8.1}$$

Here are some example results from equation 8.1 for chosen m arguments and resulting y values:

m	2^{-m}	$\sum_{n=1}^{m} 2^{-n}$	$= y$
1	$\frac{1}{2}$	$\frac{1}{2}$	$= \frac{1}{2}$
2	$\frac{1}{4}$	$\frac{1}{2} + \frac{1}{4}$	$= \frac{3}{4}$
3	$\frac{1}{8}$	$\frac{1}{2} + \frac{1}{4} + \frac{1}{8}$	$= \frac{7}{8}$
4	$\frac{1}{16}$	$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16}$	$= \frac{15}{16}$
5	$\frac{1}{32}$	$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32}$	$= \frac{31}{32}$
6	$\frac{1}{64}$	$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64}$	$= \frac{63}{64}$
7	$\frac{1}{128}$	$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} + \frac{1}{128}$	$= \frac{127}{128}$
8	$\frac{1}{256}$	$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} + \frac{1}{128} + \frac{1}{256}$	$= \frac{255}{256}$
9	$\frac{1}{512}$	$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} + \frac{1}{128} + \frac{1}{256} + \frac{1}{512}$	$= \frac{511}{512}$

Table 4: Discrete Summation

Table 4 shows that for any chosen m value, equation 8.1 will produce a sum that equals $\dfrac{2^m - 1}{2^m}$. This means a sum of 2^{-n} values can approach 1, but for finite m values can never equal it.

8.3 Infinite Pie

To show this in a more graphic way, here's a pie chart showing the result for $m = 5$ in equation 8.1:

---

[*]Remember that the operator $\sum$ takes the sum of the values generated to its right.

$$y = \sum_{n=1}^{m} 2^{-n}$$
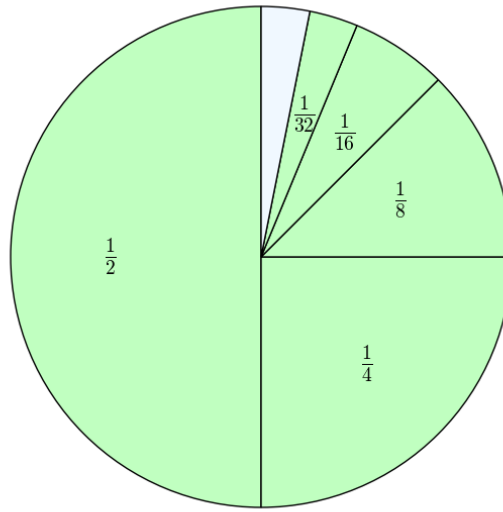
Figure 20: Discrete Summation Pie Chart

See the empty slice at the top of figure 20? Think about how we can fill that gap, so that the sum generated by equation 8.1 equals 1 – *not approximately, but exactly.*

Consider these points:

- Each new term in the equation 8.1 series adds a slice to the pie that fills half the remaining space, so regardless of the numerical value assigned to m, there will always be an unfilled gap between the sum generated by equation 8.1 and 1.
- This problem is solved by assigning to m, *not a number, but an idea* – infinity, which has this symbol: ∞. The problem is solved *precisely because* infinity is not a number, but an idea.
- Our infinity solution looks like this:

$$y = \sum_{n=1}^{\infty} 2^{-n} = 1 \tag{8.2}$$

- Remember again that infinity is an idea, not a number, and equation 8.2 produces a numerical result based on *the idea of infinity*, not a computer program that produces a sum of 2^{-n} values for n arguments between 1 and infinity.

8.4   Lots of Nines

Here's a similar problem – does this equation make a true statement?:

$$0.9999... = 1 \tag{8.3}$$

This particular equation has created much online discussion[51]. The ellipsis (...) in equation 8.3 means "Extend this sequence out to infinity" and, as with the Infinite Pie example above, it's resolved with the idea of infinity, not a number. Here's this example in summation form:

$$y = \sum_{n=1}^{m} 9 \times 10^{-n} \tag{8.4}$$

Here are some example results for equation 8.4 and finite m values:

m	9×10^{-m}	$\sum_{n=1}^{m} 9 \times 10^{-n}$	$= y$	As decimal
1	$\frac{9}{10}$	$\frac{9}{10}$	$= \frac{9}{10}$	0.9
2	$\frac{9}{100}$	$\frac{9}{10} + \frac{9}{100}$	$= \frac{99}{100}$	0.99
3	$\frac{9}{1000}$	$\frac{9}{10} + \frac{9}{100} + \frac{9}{1000}$	$= \frac{999}{1000}$	0.999
4	$\frac{9}{10000}$	$\frac{9}{10} + \frac{9}{100} + \frac{9}{1000} + \frac{9}{10000}$	$= \frac{9999}{10000}$	0.9999
5	$\frac{9}{100000}$	$\frac{9}{10} + \frac{9}{100} + \frac{9}{1000} + \frac{9}{10000} + \frac{9}{100000}$	$= \frac{99999}{100000}$	0.99999
6	$\frac{9}{1000000}$	$\frac{9}{10} + \frac{9}{100} + \frac{9}{1000} + \frac{9}{10000} + \frac{9}{100000} + \frac{9}{1000000}$	$= \frac{999999}{1000000}$	0.999999
7	$\frac{9}{10000000}$	$\frac{9}{10} + \frac{9}{100} + \frac{9}{1000} + \frac{9}{10000} + \frac{9}{100000} + \frac{9}{1000000} + \frac{9}{10000000}$	$= \frac{9999999}{10000000}$	0.9999999

Table 5: Discrete Summation 2

Each m value applied to the equation 8.4 series produces the sum $\dfrac{10^m - 1}{10^m}$, which, as before, means for finite m values the result can never equal 1. And as before, only by applying the idea of infinity can we make this sum equal 1:

$$\sum_{n=1}^{\infty} 9 \times 10^{-n} = 1 \tag{8.5}$$

So it seems this statement is true:

$$0.9999... = 1 \tag{8.6}$$

9 The Platonic Realm

9.1 Plato's Third Realm

Greek philosopher Plato[52] argued for the existence of three realms:
1. The external, physical world.
2. The internal world of conscious experience.
3. The realm of abstract forms and ideas, more clearly described in Plato's Theory of Forms/ideas[53].

In the theory of forms/ideas, Plato argued that the non-physical realm of forms and ideas was the most accurate reality, ascending above our sense perceptions and conscious perception of the external world. This concept addresses the problem of universals[54], which considers the relation between specific things and the degree to which they represent universal forms.

The Theory of Forms argues that all perceptible things are imperfect representations of universal, abstract forms and ideas, and that those abstract forms exist independent of their manifestations.

It seems to me that mathematics, even though it sometimes addresses everyday reality, resides in Plato's third realm, and many mathematical concepts represent abstract forms/ideas that sometimes have examples in physical reality. This is why, even though reality can't get along without mathematics, mathematics can easily get along with reality.

9.2 Mathematical Reality

Because of parallel developments in modern mathematics and physics, the idea is taking hold that reality is innately mathematical[55] and that the abstract realm of mathematics is the source for much of our understanding of physical reality.

Examples that support this idea appear in unexpected places. For example there's a species of cicada that hatches out at two different intervals – 13 or 17 years, no other periods. These long periods between mass births was once a mystery, but it is now thought[56] that these intervals result from the fact that both 13 and 17 are prime numbers, and because they are prime these particular reproduction intervals reduce the chance for synchronization with the

life cycles of predators. Computer mathematical models show the survival advantage of these specific intervals[57], which are thought to have arisen by the chance process of natural selection.

Modern physics has evolved in such a way that it's almost entirely described mathematically, with very little content not reliant on equations. Most modern scientific research has become a search for empirically testable equations that successfully explain past observations, make reliable predictions about observations not yet made, and more importantly provide a unifying conceptual framework that may lead to further discoveries.

In the same way, modern technology (applied science) relies on a deep understanding of mathematical ideas. For example, semiconductor technology, and the computers built from it, require an intimate understanding of quantum theory, a theory defined by mathematics.

Our ability to explore space also requires a thorough understanding of mathematics, both conceptually and in the details. A Mars-bound spacecraft was once lost because of a single wrong number[58] that resulted from confusion about which units of measurement were to be used.

9.3 Unreasonable Effectiveness

In a now-famous paper entitled "The Unreasonable Effectiveness of Mathematics in the Natural Sciences[59]," physicist Eugene Wigner ponders the surprising degree to which mathematics, which has no essential connection with reality, describes it so well. Wigner concludes his article by saying, "The miracle of the appropriateness of the language of mathematics for the formulation of the laws of physics is a wonderful gift which we neither understand nor deserve. We should be grateful for it and hope that it will remain valid in future research and that it will extend, for better or for worse, to our pleasure, even though perhaps also to our bafflement, to wide branches of learning."

It's not an exaggeration to say that understanding the modern world means understanding mathematics, and we have a simple choice – we can learn mathematics ourselves, or be perpetually lied to by those who have learned it.

10 Appendices

10.1 Data Table Design

The Acceleration/Velocity/Position data table on page 4 is meant to introduce certain calculus ideas in a simple, accessible way. Each column is the stepwise *numerical derivative* of the column to its right, and the stepwise *numerical integral* of the column to its left, where such columns exist.

10.1.1 Average Velocity

I draw the reader's attention to the "Average Velocity" column of the car data table. This column exists so that any two positions in the rightmost column will have an exact numerical derivative value available. This means that any position can be acquired by summing average velocities, and any average velocity can be acquired by subtracting two positions.

The velocity column (to the left of the average velocity column) can't be used to directly acquire positions because each stepwise change in position results from the *average* change in velocity from one position to the next, and neither the initial nor final velocity for that position can serve this purpose. Because the velocity is changing over time, the distinction between average and final velocity must be taken into account, otherwise the exact numerical correspondence between velocity and position columns would not exist.

10.1.2 No Average Acceleration

There's no average acceleration column in this table because acceleration is a constant. A strictly designed table would include an average acceleration column, but because acceleration is a constant such a column would contain the same values as the existing column, so for simplicity in design it was eliminated.

As the reader advances through this article and gets to the idea of infinitesimal steps, the distinction between average and absolute rates of change evaporates, a fact that may occur to the astute reader as another reason to learn calculus.

10.2 Detailed Symbolic Differentiation

This appendix provides a step-by-step algebraic procedure for acquiring a symbolic derivative. We begin by showing the basic strategy for moving from a function $f(x)$ to its derivative, denoted by $f'(x)$:

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{\Delta x} \tag{10.1}$$

In equation 10.1 a step variable Δx is created which (in numerical versions of this method) produces a small interval on the slope of the function's curve as shown in Figure 2 on page 8. In the symbolic strategy this variable, apart from its theoretical significance, serves to eliminate algebraic terms, in a process explained below.

10.2.1 Strategy

The basic strategy for converting a function to its derivatives relies on a combination of algebraic manipulations and removal of terms, including use of a limit operation to remove all Δx terms as well as any terms it multiplies. For this example we will use the position function first presented as equation 2.4 above:

$$p(t) = \frac{1}{2}at^2 \tag{10.2}$$

10.2.2 Sequence

We begin with this differential limit form:

$$p'(t) = \lim_{\Delta t \to 0} \frac{p(t + \Delta t) - p(t)}{\Delta t} \tag{10.3}$$

To facilitate manipulation of terms, we temporarily set aside functional notation and show the equation the function defines:

$$\frac{p(t + \Delta t) - p(t)}{\Delta t} \mapsto \frac{a(\Delta t + t)^2 - at^2}{2\,\Delta t} \tag{10.4}$$

Using the identity $(a + b)^2 = a^2 + 2ab + b^2$, we expand terms for more flexibility in term elimination:

$$\frac{a\,(\Delta t + t)^2 - at^2}{2\,\Delta t} \mapsto \frac{a\,(\Delta t^2 + 2\,\Delta t\,t + t^2) - at^2}{2\,\Delta t} \tag{10.5}$$

Solely for clarity of expression in this example, not to satisfy a mathematical requirement, we apply the factor $\frac{2}{a}$ to temporarily remove some terms:

$$\frac{a(\Delta t^2 + 2\,\Delta t\,t + t^2) - at^2}{2\,\Delta t}\,\frac{2}{a} = \frac{\Delta t^2 + 2\,\Delta t\,t + t^2 - t^2}{\Delta t} \tag{10.6}$$

We eliminate terms:

$$\frac{\Delta t^2 + 2\,\Delta t\,t + t^2 - t^2}{\Delta t} \mapsto \Delta t + 2t \tag{10.7}$$

We perform a limit operation to eliminate the remaining Δt term:

$$\lim_{\Delta t \to 0} \Delta t + 2t = 2t \tag{10.8}$$

We apply the inverse of the factor applied in equation 10.6 above:

$$2t\,\frac{a}{2} = at \tag{10.9}$$

10.2.3 Outcome

The result shown in 10.9 is the first derivative of $p(t)$, which has this notation: $p'(t)$. The above process may be concisely expressed as:

$$p'(t) = \lim_{\Delta t \to 0} \frac{a(\Delta t + t)^2 - at^2}{2\,\Delta t} = at \tag{10.10}$$

33

10.2.4 Conclusion

Notice about this process that the explicit differential approach shown in equation 10.3 above produces two avenues for transitioning from a function to its derivative: (a) algebraic elimination of terms, and (b) use of the limit operation to eliminate Δt and any terms multiplied by Δt. The order of these operations may be changed depending on circumstances.

For the nontechnical reader, the take-away from this explanation should be that a function is converted to its derivative by applying the limit operator to Δt as shown in 10.10 above, which cancels all Δt terms and any terms multiplied by them. This has the effect of transforming a function into a symbolic derivative form of that function, one that tracks *rates of change* in the original function with respect to its independent variable.

10.3 Rules for Integration

For many common equations, transformation rules can be applied to obtain integrals. Here are some examples from an old textbook[60]:

$$\int A \, dx = Ax \tag{10.11}$$

$$\int x \, dx = \frac{x^2}{2} \tag{10.12}$$

$$\int x^2 \, dx = \frac{x^3}{3} \tag{10.13}$$

$$\int x^n \, dx = \frac{x^{n+1}}{n+1}, (n \neq -1) \tag{10.14}$$

$$\int (A + Bx)^n \, dx = \frac{(A + Bx)^{n+1}}{(n+1)B}, (n \neq -1) \tag{10.15}$$

It should be clear that equation 10.14 is the general form of 10.12 and 10.13.

Here are some non-overlapping examples from a popular math website[61]:

$$\int \frac{1}{x} dx = \ln |x| \tag{10.16}$$

$$\int \frac{1}{ax + b} dx = \frac{1}{a} \ln |ax + b| \tag{10.17}$$

There are many similar sites online, these are just examples to give a sense of the notation and show the convenience of an established rule. A worthwhile general resource is Wolfram Alpha's Integral Calculator[62], which interactively accepts and solves submitted expressions.

10.4 Computer Processing Limitations

When using a computer to produce numerical differentials and because of a limitation in how computers manage numbers, beyond a certain point choosing smaller step sizes can produce a decline in accuracy:

- Typical modern computer variables[63] can resolve about 15 digits.
- Therefore if we subtract one computer number from another, if the numbers differ by more than 1×10^{15}, the result will have the value of the larger of the two numbers – it will be as if the operation never took place.
- As one approaches this limitation, the accuracy of the results declines rapidly because more and more significant digits are dropped from the calculation.

The conclusion is that we should only use numerical methods (a) to learn how these methods work, or (b) if the nature of the problem leaves us with no alternative. For the majority of problems, we should apply a symbolic differentiation method.

10.5 Calculus Notations

It turns out there are a great number of conventions to symbolize differentials and derivatives. The author uses a common notation for functions – for a given function $f(x)$, $f'(x)$ denotes first derivative, $f''(x)$ second derivative and so forth – this so-called "prime" notation originated with Lagrange[64]. One drawback to this convention is that it doesn't explicitly distinguish dependent and independent variables (the function argument is assumed to represent the independent variable). For purposes other than functional notation, where both dependent and independent variables are present, Gottfried Liebniz's derivative notation[65] might be preferable – for independent and dependent variables x and y:

$$\frac{dy}{dx}(\text{first derivative}) , \frac{d^2y}{dx^2}(\text{second derivative}), \text{etc.} \tag{10.18}$$

Liebniz's derivative notation for functions has a number of variations, some of which are:

$$\frac{df}{dx}(x) \text{ or } \frac{df(x)}{dx} \text{ or } \frac{d}{dx}f(x) \tag{10.19}$$

But it's not clear that this has an advantage over the Lagrange "prime" notation. There are many other notations in current use and with historical importance[66].

References

[1]Accelerometer – a device for measuring changes in velocity.

[2]Inverse (mathematics) – two functions or operations that perform opposite actions are said to be *inverses* of each other.

[3]Derivative (mathematics) – "The derivative of a function of a real variable measures the sensitivity to change of a quantity (a function value or dependent variable) which is determined by another quantity (the independent variable)."

[4]Integral (mathematics) – "In mathematics, an integral assigns numbers to functions in a way that can describe displacement, area, volume, and other concepts that arise by combining infinitesimal data."

[5]Fundamental theorem of calculus – The idea that differentiation and integration are inverse operations.

[6]Continuous function – roughly speaking, a function whose output smoothly and continuously follows its input, with no discontinuities or interruptions.

[7]Variable (mathematics) – a symbol that stands for a numerical quantity.

[8]Independent variable – a variable that doesn't result from a computation and whose value doesn't depend on another variable.

[9]Dependent variable – a variable whose value depends on the outcome of a mathematical operation that in turn depends on the value of an independent variable.

[10]Mathematical constant – an unchanging numerical value.

[11]Function (mathematics) – a function is a symbol stands for an equation.

[12]Tangent Line – a line that intersects a function's plot line and matches its slope as well.

[13]Division by zero – an undefined mathematical operation.

[14]Limit (mathematics) – a method to approach values that cannot be directly assigned.

[15]Error function – a very widely used function that cannot be integrated symbolically.

[16]Three-body problem (orbital mechanics) – the class of orbital problems lacking closed-form solutions.

[17]Definite Integral – definition in terms of indefinite integrals.

[18]Integration as summation (PDF) – a concise summary showing use of the limit operator to show the equivalence of a particular kind of summation and an integral.

[19]Antiderivative – the opposite of a derivative, an integral.

[20]Storage Container Mathematics – mathematics applied to storage vessels.

[21]Differential equation – an equation that relates a function to its derivatives.

[22]First-Order Ordinary Differential Equation

[23]Initial value problem – a differential equation that includes one or more initial values.

[24]Natural logarithm

[25]The Mathematics of Population Increase – differential equations applied to measurement of population growth.

[26]Computer algebra system – a class of computer software able to perform relatively high-level mathematics.

[27]Python (programming language) – a very popular programming language with libraries able to perform numerical and symbolic mathematics.

[28]SymPy – a Python library for symbolic computation.

[29]Jupyter – an interactive mathematical environment.

[30]ISymPy Console – a command-shell-based SymPy environment.

[31]SymPy Online Shell – an interactive ISymPy command-line environment in a Web page.

[32]IPython kernels for other languages – essentially, Jupyter supported kernels.

[33]SageMath – a free, open-source CAS based on Python.

[34]Exploring Mathematics with Sage – the author's Sage tutorial.

[35]Maxima – a CAS based on an earlier project named Macsyma and used in SageMath for symbolic results.

[36]Wolfram Mathematica – a.k.a. Mathematica, a powerful, and powerfully expensive, CAS.

[37]Gravity of Earth – also known as little-g, approximately 9.8 $m/s/s$

[38]Jerk (physics) – The semi-official name for the third derivative of position.

[39]Raspbian – a Linux distribution designed for the Raspberry Pi computers.

[40]Raspberry Pi – a series of small, inexpensive but surprisingly powerful computers.

[41]Wolfram – the source of Mathematica.

[42]Anaconda Download – the download page for all versions of Anaconda.

[43]Virtual machine – a scheme in which what is essentially a computer in software is hosted on a different machine.

[44]SageAPpliance – a package that hosts SageMath on any target machine by way of a virtual machine.

[45]VirtualBox – a popular cross-platform virtual machine manager/hosting environment.

[46]Conservation of energy – a physical theory in which energy is neither created nor destroyed, only changed in form

[47]Kinetic energy – the energy of motion.

[48]Gravitational potential energy – an energy that is proportional to position.

[49]Gravitational constant – an important physical constant having to do with gravitation.

[50]Kepler's laws of planetary motion – an early effort to systematize our understanding of planetary orbits.

[51]0.999... – a discussion of the identity 0.999... = 1

[52]Plato – influential Greek philosopher.

[53]Theory of Forms – a Platonic realm of ideas and abstract forms.

[54]Problem of universals – the philosophical problem addressing the relation between specific things and what abstract forms they represent

[55]Mathematical universe hypothesis – the hypothesis that the universe is innately mathematical.

[56]Periodical cicadas – a species of cicada that, for survival reasons, reproduce at prime-numbered intervals.

[57]Mathematical Locusts – how mathematics plays a part in species survival.

[58]Mars Climate Orbiter – a spacecraft lost due to one wrong number.

[59]The Unreasonable Effectiveness of Mathematics in the Natural Sciences –an article that discusses the power of mathematics to describe reality.

[60]These examples come from "A New Table of Indefinite Integrals – Computer Processed", 1971, Melvin Klerer & Fred Grossman, Dover, ISBN 0-486-62714-4.

[61]Table of Basic Integrals – a compendium of conversion rules for indefinite integrals.

[62]Online Integral Calculator – A subset of Wolfram Alpha's online offering that calculates definite and indefinite integrals.

[63]IEEE 754: floating point in modern computers – a description of a typical modern computer numerical variable.

[64]Lagrange's Notation – the "prime" notation for function derivatives used in this article.

[65]Leibniz's notation – mathematical notation for derivatives.

[66]Notation for Differentiation – a summary of many notations, current and historical.