1.0 Newton's Laws

Some 320 years ago, Isaac Newton wrote down the two basic equations of classical Physics

the basic equation of force $F = m\frac{dv}{dt} = m\dot{v}$

and the force of gravity $m\,r = -\frac{mGM}{r^2}$

Let us begin with motion of a planet in a gravity field. We'll use the metric system to describe the motion of the Earth about the Sun.

Cancelling m, $r = -\frac{GM}{r^2}$ where $M = M_{sun} = 1.989 \times 10^{30}$ kg

and $G = 6.673 \times 10^{-11} \frac{m^3}{kg \cdot s^2}$

Notice that if we wish to measure distance in km and time in hours, then we need only change the units of G

$$G = 6.673 \times 10^{-11} \frac{m^3}{kg \cdot s^2} = 6.673 \times 10^{-20} \frac{km^3}{kg \cdot s^2}$$

$$G = 3600^2 \times 6.673 \times 10^{-20} \frac{km^3}{kg \cdot (hr)^2}$$

G is exactly the same. Only the units are changed. Choose the G you need, and express your distance units in m or km and your time in s or hr, but <u>be consistent</u> throughout your equation.

Now, back to the Earth. If we place the Earth at its *average distance* $r = 149.6 \times 10^6$ km away from the Sun, in a circular orbit with velocity v,

$$\frac{mv^2}{r} = \frac{mGM}{r^2}$$

$$v = \sqrt{\frac{GM}{r}} = 29.786 \text{ km/s}$$

Let's now write the physics equations for *circular motion* of the Earth about the Sun. To begin, we'll use x-y coordinates.

$$\ddot{x} = -\frac{GM}{x^2+y^2}\cos\theta = -\frac{GM\,x}{(x^2+y^2)^{3/2}} \qquad x_o = 149.6\times 10^6 \text{ km} \qquad \dot{x}_o = 0$$

$$\ddot{y} = -\frac{GM}{x^2+y^2}\sin\theta = -\frac{GM\,y}{(x^2+y^2)^{3/2}} \qquad y_o = 0 \qquad \dot{y}_o = 29.786 \text{ km/s}$$

These two equations describe the motion of the Earth for all time, based on the initial conditions. Let's look at the requirements for finding the speed and location of the Earth at all times after $t = 0$. First of all these two equations are complicated and they are not linear. We shall find a solution by using numerical methods (**NDSolve** in *Mathematica*). We will solve from $t = 0$ to $t = 9000$ hours.

Here is how we do it. To get the time in hours, and the distance from the sun in kilometers, take

$$G = 3600^2 \times 6.673 \times 10^{-20} \frac{\text{km}^3}{\text{kg}\cdot(\text{hr})^2} \quad \text{and} \quad M_{sun} = 1.989\times 10^{30} \text{ kg}$$

Then the x- and y-distance of the Earth from the center of the Sun will be measured in kilometers. Notice that to be consistent we will also have to convert the initial velocity into kilometers per hour.

Here is the *Mathematica* program

```
In[1]:=  G = 3600^2 * 6.673 * 10^-20; M = 1.989 * 10^30;
         sol = NDSolve[{
            x''[t] == G * M * (-x[t]) / (x[t]^2 + y[t]^2)^1.5,
            y''[t] == G * M * (-y[t]) / (x[t]^2 + y[t]^2)^1.5,
            x[0] == 149.6 * 10^6, y[0] == 0, x'[0] == 0, y'[0] == 29.786 * 3600},
           {x, y}, {t, 0, 9000}]
```

When we press **Shift-Enter**, the *Mathematica* program returns

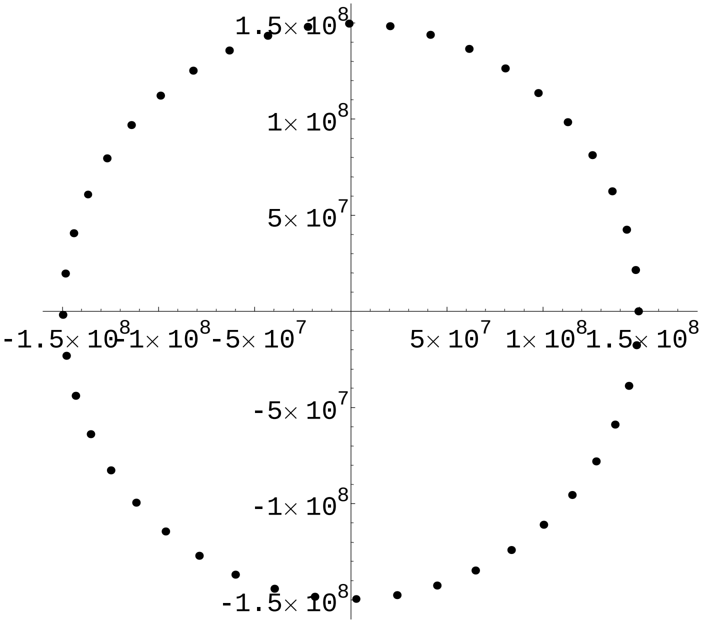
```
Out[1]=  {{x → InterpolatingFunction[{{0., 9000.}}, <>],
           y → InterpolatingFunction[{{0., 9000.}}, <>]}}
```

x and y are given by an Interpolating Function. We can plot x vs y if we first identify an interpolation function in x, and an interpolation function in y and table these values versus time.

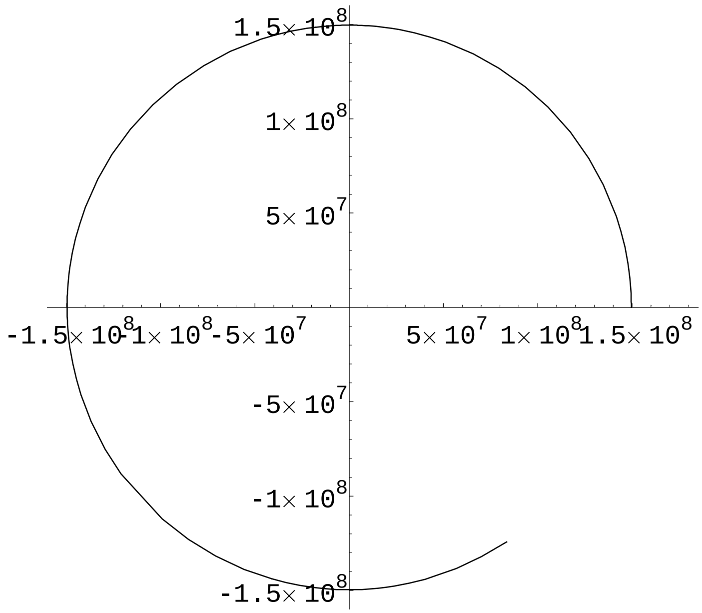
```
InterpFunc1 = x /. sol[[1]]; InterpFunc2 = y /. sol[[1]];
InterpFunc3 = x′ /. sol[[1]]; InterpFunc4 = y′ /. sol[[1]];
tbl = Table[{InterpFunc1[t], InterpFunc2[t]}, {t, 0, 8600, 200}];
```

Now we are free to plot a series of (`x, y`) points versus time. Either as a series of dots (**`ListPlot`**) or as a continuous line (**`ParametricPlot`**).

```
ListPlot[tbl, AspectRatio → Automatic, Prolog → AbsolutePointSize[3]]
```




```
ParametricPlot[{x[t], y[t]} /. sol, {t, 0, 7400}, AspectRatio → Automatic]
```



A Parametric Plot of the Earth in a circular orbit.

Let us now **`Table`** the data from the Interpolation Functions.

```
In[191]:= r[t_] = √(InterpFunc1[t]^2 + InterpFunc2[t]^2);
          v[t_] = √(InterpFunc3[t]^2 + InterpFunc4[t]^2) / 3600;
          Table[{t, InterpFunc1[t], InterpFunc2[t], r[t], v[t]},
            {t, 0, 8800, 400}] // TableForm
```

t (hr)	x (km)	y (km)	r (km)	v (km / s)
0	1.496×10^{8}	-1.43599×10^{-21}	1.496×10^{8}	29.786
400	1.43493×10^{8}	4.23066×10^{7}	1.496×10^{8}	29.786
800	1.25672×10^{8}	8.11593×10^{7}	1.496×10^{8}	29.786
1200	9.75899×10^{7}	1.13386×10^{8}	1.496×10^{8}	29.786
1600	6.15408×10^{7}	1.36356×10^{8}	1.496×10^{8}	29.786
2000	2.04676×10^{7}	1.48193×10^{8}	1.496×10^{8}	29.786
2400	-2.22767×10^{7}	1.47932×10^{8}	1.496×10^{8}	29.786
2800	-6.32023×10^{7}	1.35594×10^{8}	1.496×10^{8}	29.786
3200	-9.8968×10^{7}	1.12185×10^{8}	1.496×10^{8}	29.786
3600	-1.26654×10^{8}	7.96178×10^{7}	1.496×10^{8}	29.786
4000	-1.43999×10^{8}	4.05503×10^{7}	1.496×10^{8}	29.786
4400	-1.49589×10^{8}	-1.82776×10^{6}	1.496×10^{8}	29.786
4800	-1.42966×10^{8}	-4.40566×10^{7}	1.496×10^{8}	29.786
5200	-1.24671×10^{8}	-8.26886×10^{7}	1.496×10^{8}	29.786
5600	-9.61974×10^{7}	-1.1457×10^{8}	1.496×10^{8}	29.786
6000	-5.98704×10^{7}	-1.37097×10^{8}	1.496×10^{8}	29.786
6400	-1.86556×10^{7}	-1.48432×10^{8}	1.496×10^{8}	29.786
6800	2.40823×10^{7}	-1.47649×10^{8}	1.496×10^{8}	29.786
7200	6.48541×10^{7}	-1.34811×10^{8}	1.496×10^{8}	29.786
7600	1.00331×10^{8}	-1.10968×10^{8}	1.496×10^{8}	29.786
8000	1.27617×10^{8}	-7.80645×10^{7}	1.496×10^{8}	29.786
8400	1.44484×10^{8}	-3.8788×10^{7}	1.496×10^{8}	29.786
8800	1.49555×10^{8}	3.65522×10^{6}	1.496×10^{8}	29.786

The above Table shows the Earth in a circular orbit about the Sun. For our choice of starting conditions, `r` and `v` remain constant.

Just how accurate is *Mathematica*? Notice that in the above **`Table`** that we have solved the Differential Equations for a time of one year, and, as expected, the radius of the orbit and the velocity have remained absolutely constant. This is a first test of the *Mathematica* numerical solver, and it is gratifying to see that the LSODA algorithm in **`NDSolve`** produces good, consistent results. (The Livermore Solver for Ordinary Differential equations Adaptive method was developed at Lawrence Livermore Labs and utilizes both an Adams method and a Gear backward differences method to obtain results of high accuracy.)

In terms of precision, *Mathematica* routinely carries at least 16 places of decimal accuracy. This is far more than what we will need in this text, where we shall report the results of computations to 6 places (1 part per million accuracy) except in those few cases where greater precision is required.

To this point, we have only modeled the Earth as traveling in a circular orbit at one Astronomical Unit (its average distance of 149.6 million kilometers) from the center of the sun. What is necessary next, is to use the computer to find all the parameters of the Earth's *elliptical* orbit about the sun, using only Newton's Law of Gravity, and a starting distance and velocity for the Earth.