Moving in Three Dimensions

Introduction

OBJECTIVE: Learn to use *Mathematica* to perform calculations to analyze motion in the equations that are given parametrically.

Moving in three dimensions is something with which we are familiar, but visualization of equations of motion and computations involved in analyzing that motion can be cumbersome. This module gives a broad overview on the analysis of motion where the equations are given parametrically.

▪ Technology Guidelines

NOTE: If you have just finished a module, restart *Mathematica* or close the *Kernel* before executing a new module.
TO OPEN CELLS, put your cursor on the right cell bracket and double click.
TO STOP AN EXECUTION
Select the *Kernel* pull-down menu and click on *Abort Evaluation.*
ORDER OF EXECUTION
Execute cells in the order given. Do not skip any Input cells within a given notebook.
SAVING NOTEBOOKS
You can save anytime to any directory you choose, and it is wise to save often.
However, before you do your final save, it is a good idea to delete all your output by selecting the
Delete All Output selection under the *Kernel* pull-down menu.
EXPERIENCING MAJOR PROBLEMS
Save if appropriate, then shut down *Mathematica* and start it up again.

Part I: Parametric Equations of a Curve in 3-Space

First, we define the *x*, *y*, and *z* coordinates for motion parametrically.

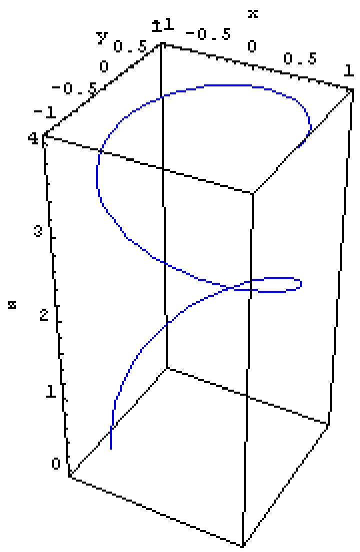
In[1]:=

```
Off[General::spell]

Off[General::spell1]

Clear[x, y, z, t]

x[t_] := Cos[t]

y[t_] := Sin[t]

z[t_] := 4 - t^2 / 25
```

Next, we plot the resulting curve in red. Can you tell in which direction you are moving on the curve as *t* increases?

In[7]:=

```
plotf = ParametricPlot3D[{x[t], y[t], z[t], R(
  {t, 0, 10}, AxesLabel -> {"x", "y", "z"}];
```



If you consider the parametric equation as a vector equation for the motion of a particle, the derivative of that vector is the velocity vector that we form by differentiating each component. The commands below plot both the first and second derivative of the vector position function.

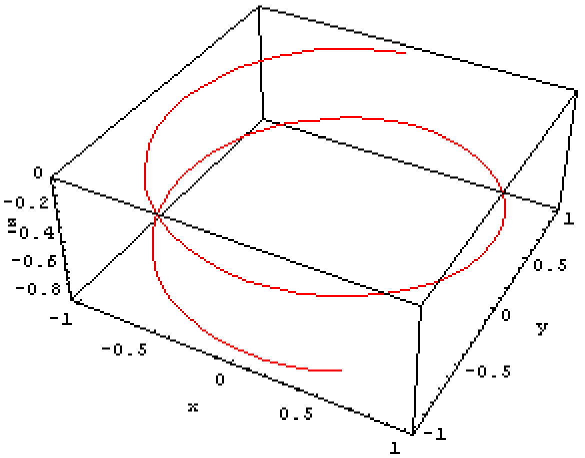
"> About *Mathematica*

In[8]:=

```
Print["The velocity vector is ", {x'[t], y'
plotf' = ParametricPlot3D[
   {x'[t], y'[t], z'[t], RGBColor[1, 0, 0]}, 
   AxesLabel -> {x, y, z}];
Print["The acceleration vector is ",
  {x''[t], y''[t], z''[t]}]
plotf'' = ParametricPlot3D[
   {x''[t], y''[t], z''[t], RGBColor[0, 1, 1
   AxesLabel -> {x, y, z}];
```

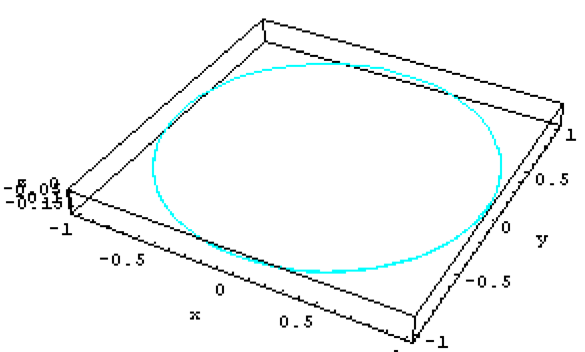
The velocity vector is

$\left\{-\text{Sin}[t], \text{Cos}[t], -\frac{2t}{25}\right\}$



The acceleration vector is

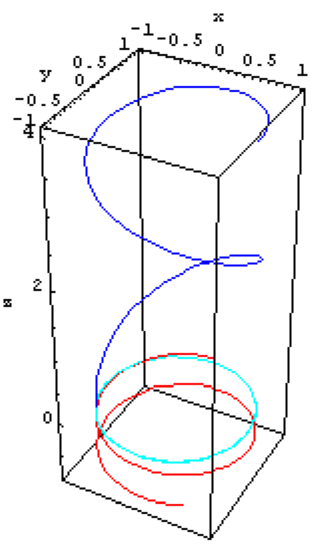
$\left\{-\text{Cos}[t], -\text{Sin}[t], -\frac{2}{25}\right\}$



The following command plots all three functions together, with the position in blue, the velocity in red, and the acceleration in aqua. You can see that the acceleration in the vertical direction is a negative constant, so the velocity in that direction is decreasing at a linear rate, causing the motion to spiral down at a rate proportion to t^2. The motion, velocity, and acceleration in the *x* and *y* directions are circular.

In[12]:=

```
Show[plotf, plotf', plotf''];
```



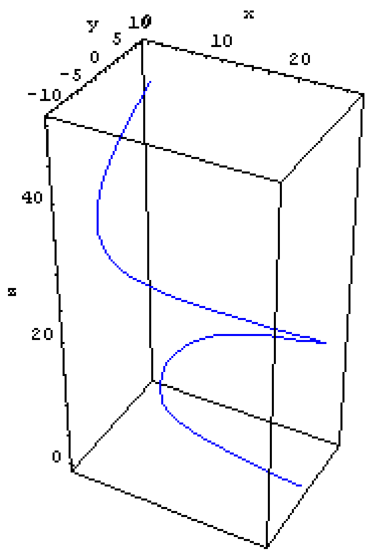
You Try It: Part I

Define your own *x*, *y*, and *z* coordinates for motion by changing the entries in red. You may or may not want to change the time interval (in red) over which you plot your functions.

In[13]:=

```
Clear[x, y, z, t]

x[t_] := 10 Exp[Cos[t]]

y[t_] := 10 Sin[t^2 / 15]

z[t_] := t^2 / 2
```

```
timeinterval = 5;
plotf = ParametricPlot3D[{x[t], y[t], z[t], RG
  {t, 0, 10}, AxesLabel -> {"x", "y", "z"}];
```



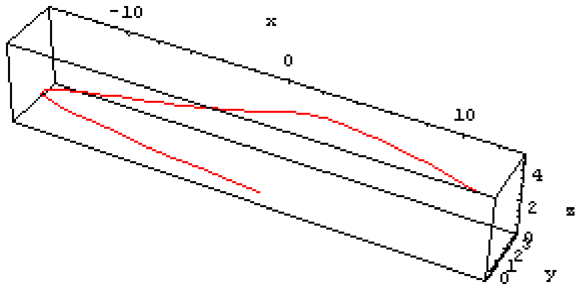
Now check out the velocity and acceleration.

In[18]:=

```
Print["The velocity vector is ", {x'[t], y'[

plotf' = ParametricPlot3D[
  {x'[t], y'[t], z'[t], RGBColor[1, 0, 0]},
  {t, 0, timeinterval}, AxesLabel -> {x, y, z}]

Print["The acceleration vector is ", {x''[t]

plotf'' = ParametricPlot3D[
  {x''[t], y''[t], z''[t], RGBColor[0, 1, 1]
  {t, 0, timeinterval}, AxesLabel -> {x, y, z}]

Show[plotf, plotf', plotf''];
```

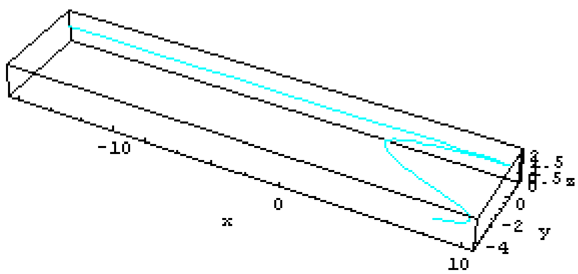
The velocity vector is

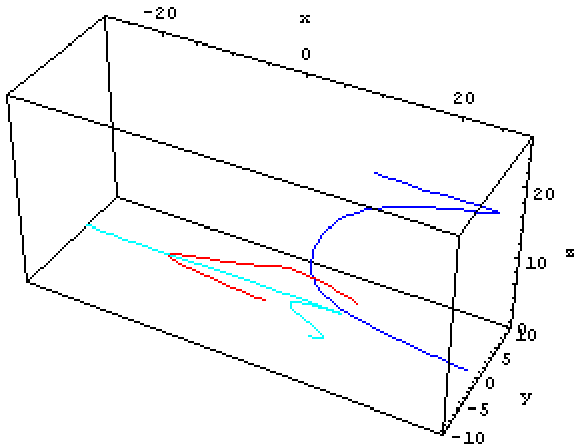
$$\left\{-10\, e^{\mathrm{Cos}[t]}\, \mathrm{Sin}[t],\ \frac{4}{3}\, t\, \mathrm{Cos}\left[\frac{t^2}{15}\right],\ t\right\}$$



The acceleration vector is

$$\left\{-10\, e^{\text{Cos}[t]}\, \text{Cos}[t] + 10\, e^{\text{Cos}[t]}\, \text{Sin}[t]^2,\ \frac{4}{3}\, \text{Cos}\left[\frac{t^2}{15}\right] - \frac{8}{45}\, t^2\, \text{Sin}\left[\frac{t^2}{15}\right],\ 1\right\}$$





The plot of all three together may or may not be instructive.

Compare your results to those of your classmates.

Part II: Equations of Motion
Velocity → Position

Velocity → Acceleration

If you are given the velocity, you can differentiate to find the acceleration and integrate to find the displacement. Since we do three integrations, we need three arbitrary constants to get the general solution.

In[23]:=

```
Clear[position, velocity, acceleration, t, a

Print["The velocity vector is defined to be
 velocity = {3 / 2 (t + 1)^1/2, E^-t, 1 / (t + 1)}]

Print["The acceleration vector is ",
 acceleration = D[velocity, t]]

Print["The position vector is ",
 rgeneral = Integrate[velocity, t] + {a, b, c}]
```

The velocity vector is defined to be

$$\left\{\frac{3\sqrt{1+t}}{2}, e^{-t}, \frac{1}{1+t}\right\}$$

The acceleration vector is

$$\left\{\frac{3}{4\sqrt{1+t}}, -e^{-t}, -\frac{1}{(1+t)^2}\right\}$$

The position vector is

$$\{a+(1+t)^{3/2}, b-e^{-t}, c+\text{Log}[1+t]\}$$

To evaluate the constants of integration in the formula for displacement (**rgeneral**), we need to apply the initial conditions. The following command computes those constants for an initial position of (1,1,0).

In[27]:=

```
initial = {1, 1, 0};

constants = Solve[(rgeneral /. t -> 0) == initia
```

Out[28]=

```
{{a → 0, b → 2, c → 0}}
```

We evaluate our position vector with the constants replaced by the values just determined.

In[29]:=

```
r = rgeneral /. constants
```

Out[29]=

$$\{\{(1+t)^{3/2}, 2-e^{-t}, \text{Log}[1+t]\}\}$$

You Try It: Part II

Change the entries in your velocity vector and your initial position (items in red), and re-execute the cell.

In[30]:=

```
Clear[a, b, c, t]

velocity = {3/2 (t + 1)^1/2, e^-t, 1/(t + 1)}

acceleration = D[velocity, t]

rgeneral = Integrate[velocity, t] + {a, b, c}

initial = {1, 1, 0};

constants = Solve[(rgeneral /. t -> 0) == initia

r = rgeneral /. constants
```

Out[31]=

$$\left\{\frac{3\sqrt{1+t}}{2}, e^{-t}, \frac{1}{1+t}\right\}$$

Out[32]=

$$\left\{\frac{3}{4\sqrt{1+t}}, -e^{-t}, -\frac{1}{(1+t)^2}\right\}$$

Out[33]=

```
{a + (1 + t)^(3/2), b - e^(-t), c + Log[1 + t]}
```

Out[35]=

```
{{a → 0, b → 2, c → 0}}
```

Out[36]=

```
{{(1 + t)^(3/2), 2 - e^(-t), Log[1 + t]}}
```

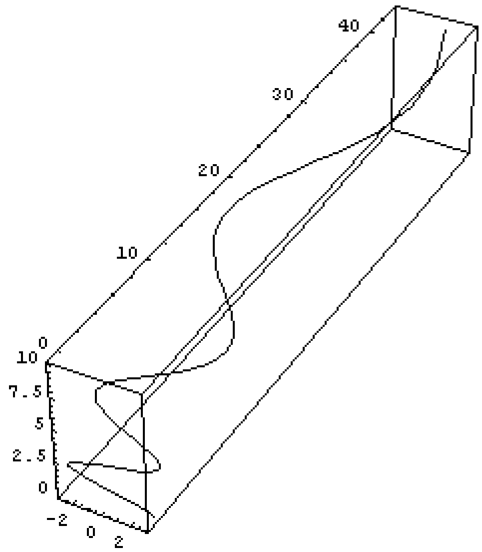
Part III: Computing the Distance Traveled on a Curved Path

Suppose that you are walking up a mountain along the path given below. You can use the following code to compute the distance traveled in the first ten minutes, where *t* stands for time in minutes and distance is measured in feet.

In[37]:=

```
Clear[r, t, v]

Print["The postion given in feet is ",
 r[t_] = {3 Cos[2 t], t^(10/3)/50, t}];

ParametricPlot3D[Evaluate[r[t]], {t, 0, 10}];

Print["The velocity vector given in feet pe
 v[t_] = D[r[t], t]]

Print["The speed (ds/dt) in feet per minute i
  speed = √(v[t].v[t]) ] // Simplify

Print["The distance traveled in feet in 10 
 distance = NIntegrate[speed, {t, 0, 10}]]
```

```
The postion given in feet is
 {3 Cos[2 t], t^(10/3)/50, t}
```



The velocity vector given in feet
per minute is $\left\{-6\,\mathrm{Sin}[2\,t],\ \frac{t^{7/3}}{15},\ 1\right\}$

The speed $(\frac{ds}{dt})$ in feet per minute is
$\sqrt{1+\frac{t^{14/3}}{225}+36\,\mathrm{Sin}[2\,t]^2}$

The distance traveled
in feet in 10 minutes is 66.027

In the above example, if you had tried to integrate symbolically, instead of numerically, what would have happened? (The **N** in front of the **Integrate** signifies numerical integration.) Many integrals arising in the computation of arc length are too complicated to evaluate symbolically.

In[43]:=

```
Integrate[speed, {t, 0, 10}]
```

Out[43]=

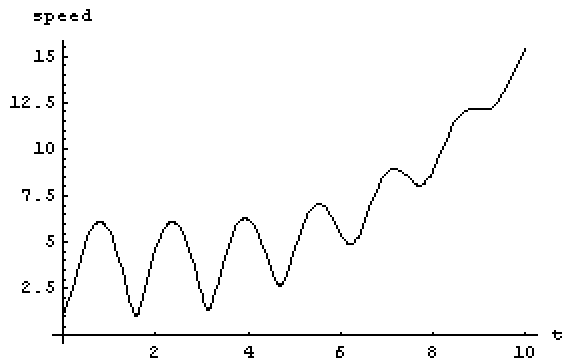
$$\int_0^{10}\sqrt{1+\frac{t^{14/3}}{225}+36\,\mathrm{Sin}[2\,t]^2}\,dt$$

Check your speed after five minutes and plot your speed during the first ten minutes.

In[44]:=

```
Print["speed at 5 minutes is ", speed /. t → 5
 " feet per minute"]

Plot[speed, {t, 0, 10}, AxesLabel -> {t, "speed
```

```
speed at 5 minutes is
 4.44711 feet per minute
```



If *t* represents time in minutes, what does this say about the speed of the hiker? Is it possible?

You Try It: Part III

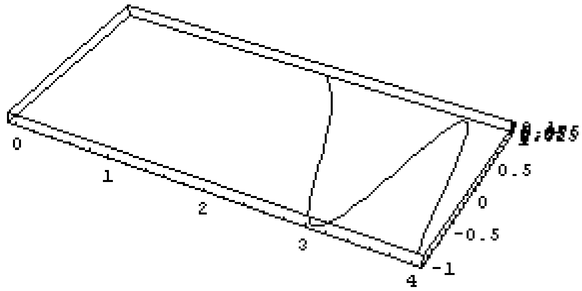
Choose your own position function and the interval over which you wish to integrate (terms in red).

In[46]:=

```
Clear[r, t, v]
Print["The postion vector given in feet is "
  r[t_] = {2 t^3, Cos[t - 1], Exp[-t^2]}];
 ParametricPlot3D[Evaluate[r[t]], {t, 0, 10}]
Print["The velocity given in feet per minute
  v[t_] = D[r[t], t]]
Print["The speed (ds/dt) in feet per minute is
   speed = √(v[t].v[t]) ] // Simplify
Print["The distance traveled in feet in the
    minutes is ", distance = NIntegrate[speed
```

```
The postion vector given in feet is
```

$\{2\,t^{0.3},\ \mathrm{Cos}[1-t],\ e^{-t^2}\}$




```
The velocity given in feet per minute is
```

$\left\{\frac{0.6}{t^{0.7}},\ \mathrm{Sin}[1-t],\ -2\,e^{-t^2}\,t\right\}$

The speed $(\frac{ds}{dt})$ in feet per minute is

$\sqrt{\frac{0.36}{t^{1.4}} + 4\,e^{-2\,t^2}\,t^2 + \mathrm{Sin}[1-t]^2}$

```
The distance traveled in feet
   in the first 10 minutes is 8.8625
```

Out[46]=

Null^4

Part IV: Computing Curvature and Torsion for a Space Curve

Mathematica simplifies the process of finding curvature and torsion. The computations below use formulas directly from your text, and we graphically explore the interpretations of the curvature and torsion functions.

In[47]:=

```
Clear[r, v, t, speed]
```

$\mathrm{mag[vector\_] := \sqrt{vector.vector}}$

```
Print["The position vector is ",
 r[t_] = {10 Sin[t], 10 Exp[-t], t^(10/3)/100}]

Print["The velocity vector is ", v[t_] = r'[t

Print["The acceleration vector is ", a[t_] =

Print["The speed is ", speed[t_] = mag[v[t]]]

Print["The unit tangent vector is ",
 utan[t_] = v[t] / speed[t] // Simplify]

Print["The curvature is ",
 curvature[t_] = mag[Cross[v[t], a[t]]] / spee

Print["The torsion is ",
 torsion[t_] =
  Det[{v[t], a[t], a'[t]}] /
    (Cross[v[t], a[t]].Cross[v[t], a[t]]) // S
```

The position vector is $\left\{10\,\mathrm{Sin}[t],\ 10\,e^{-t},\ \frac{t^{10/3}}{100}\right\}$

The velocity vector is $\left\{10\,\mathrm{Cos}[t],\ -10\,e^{-t},\ \frac{t^{7/3}}{30}\right\}$

The acceleration vector is $\left\{-10\,\mathrm{Sin}[t],\ 10\,e^{-t},\ \frac{7\,t^{4/3}}{90}\right\}$

The speed is $\sqrt{100\,e^{-2t} + \frac{t^{14/3}}{900} + 100\,\mathrm{Cos}[t]^2}$

The unit tangent vector is

$$\left\{\frac{300\,\mathrm{Cos}[t]}{\sqrt{90000\,e^{-2t}+t^{14/3}+90000\,\mathrm{Cos}[t]^2}},\ -\frac{300\,e^{-t}}{\sqrt{90000\,e^{-2t}+t^{14/3}+90000\,\mathrm{Cos}[t]^2}},\ \frac{t^{7/3}}{\sqrt{90000\,e^{-2t}+t^{14/3}+90000\,\mathrm{Cos}[t]^2}}\right\}$$

The curvature is

$$\left(3000\sqrt{\left(e^{-2t}\,t^{8/3}\,(7+3\,t)^2+t^{8/3}\,(7\,\mathrm{Cos}[t]+3\,t\,\mathrm{Sin}[t])^2+810000\,e^{-2t}\,(1-2\,\mathrm{Cos}[t]\,\mathrm{Sin}[t])\right)}\right)\Big/\left(90000\,e^{-2t}+t^{14/3}+90000\,\mathrm{Cos}[t]^2\right)^{3/2}$$

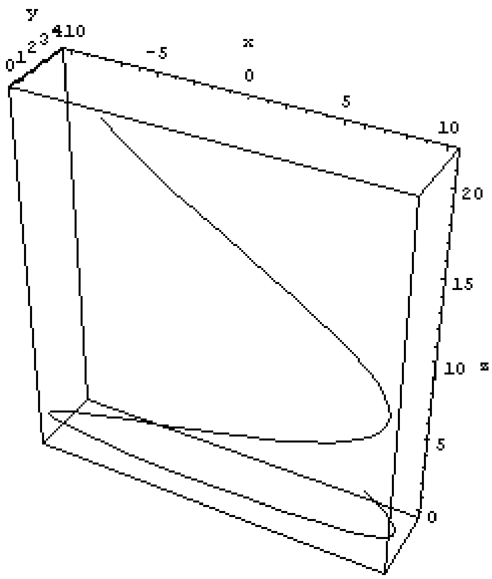
The torsion is

$$\left(30\,e^{t}\,t^{1/3}\,\left((28+42\,t+9\,t^2)\,\mathrm{Cos}[t]+(-28+9\,t^2)\,\mathrm{Sin}[t]\right)\right)\Big/\left(49\,t^{8/3}+42\,t^{11/3}+9\,t^{14/3}+(810000+49\,e^{2t}\,t^{8/3})\,\mathrm{Cos}[t]^2+9\,(90000+e^{2t}\,t^{14/3})\,\mathrm{Sin}[t]^2-810000\,\mathrm{Sin}[2\,t]+21\,e^{2t}\,t^{11/3}\,\mathrm{Sin}[2\,t]\right)$$

Let's visualize some of these quantities. We will begin by drawing the path of motion.

In[56]:=

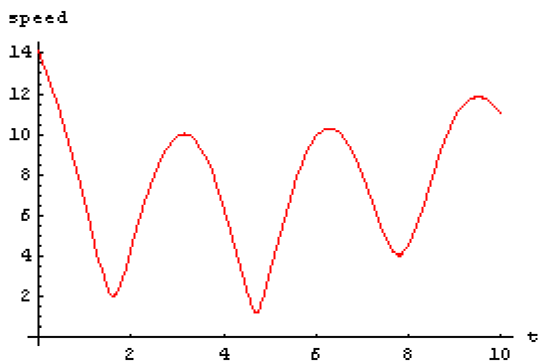
```
ParametricPlot3D[Evaluate[r[t]], {t, 0, 10},
 AxesLabel -> {x, y, z}];
```



In[57]:=

```
Print["The speed is ", speed[t]]

ps = Plot[Evaluate[speed[t]], {t, 0, 10}, Axes
  PlotStyle -> RGBColor[1, 0, 0]];
```

The speed is $\sqrt{100\, e^{-2t} + \frac{t^{14/3}}{900} + 100\,\mathrm{Cos}[t]^2}$

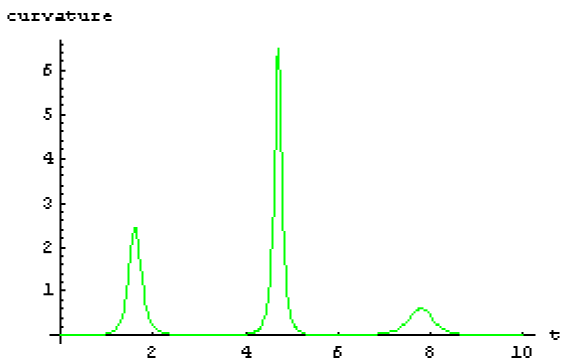


In[59]:=

```
Print["The curvature is ", curvature[t]]

pc = Plot[Evaluate[curvature[t]], {t, 0, 10},
  AxesLabel -> {t, curvature}, PlotRange -> All
  PlotStyle -> RGBColor[0, 1, 0]];
```

The curvature is

$$\left(3000\sqrt{\left(e^{-2t}\,t^{8/3}\,(7+3t)^2 + t^{8/3}\,(7\,\mathrm{Cos}[t] + 3\,t\,\mathrm{Sin}[t])^2 + 810000\,e^{-2t}\,(1 - 2\,\mathrm{Cos}[t]\,\mathrm{Sin}[t])\right)}\right) \Big/ \left(90000\,e^{-2t} + t^{14/3} + 90000\,\mathrm{Cos}[t]^2\right)^{3/2}$$

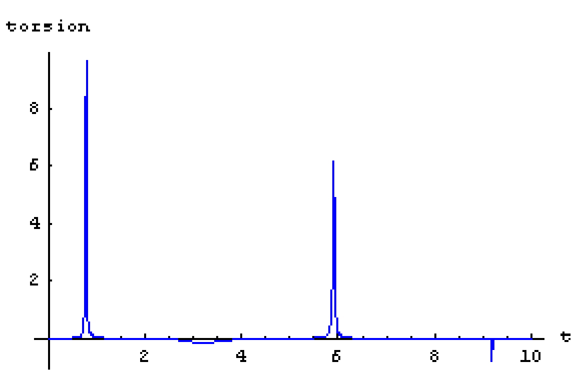


In[61]:=

```
Print["The torsion is ", torsion[t]]

pt = Plot[Evaluate[torsion[t]], {t, 0, 10},
  AxesLabel -> {t, torsion}, PlotStyle -> RGBC
  PlotRange -> All];
```

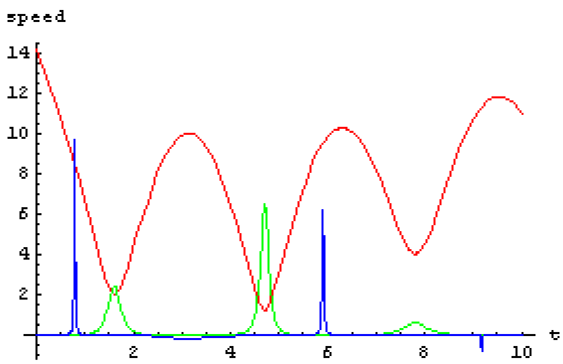
The torsion is

$$\left(30\,e^{t}\,t^{1/3}\,\left((28 + 42\,t + 9\,t^2)\,\mathrm{Cos}[t] + (-28 + 9\,t^2)\,\mathrm{Sin}[t]\right)\right) \Big/ \left(49\,t^{8/3} + 42\,t^{11/3} + 9\,t^{14/3} + (810000 + 49\,e^{2t}\,t^{8/3})\,\mathrm{Cos}[t]^2 + 9\,(90000 + e^{2t}\,t^{14/3})\,\mathrm{Sin}[t]^2 - 810000\,\mathrm{Sin}[2\,t] + 21\,e^{2t}\,t^{11/3}\,\mathrm{Sin}[2\,t]\right)$$



In[63]:=

```
Show[ps, pc, pt];

Print["speed in red, curvature in green, tor
```




```
speed in red, curvature
  in green, torsion in blue
```

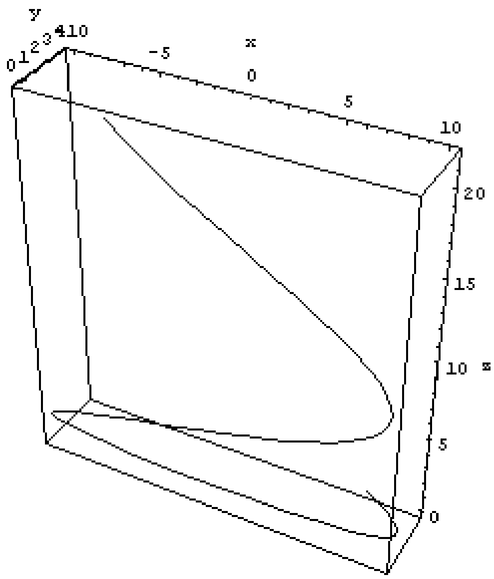
Look at your space curve, and see if you can identify the points on your space curve at which the curvature or torsion spike. What has caused these events? Are they related at all to the speed?

From the definition of curvature that relates it to the cross product of the velocity and acceleration, recall that the curvature is smallest when the velocity and acceleration are in the same direction. The curvature is largest when the velocity and acceleration are perpendicular to one another.

Consider torsion as the dot product of the derivative of the unit binormal and the normal vectors. Torsion will peak when those two vectors are in the same direction and will be close to 0 when those two vectors are perpendicular.

In[65]:=

```
ParametricPlot3D[Evaluate[r[t]], {t, 0, 10},
 AxesLabel -> {x, y, z}];
```



See the TNB frame for a curve in action in Part VI of this module.

You Try It: Part IV

Redefine your position function (in red) and compute it. Note that the computations here are not trivial, so complicated functions may take some time to compute.

In[66]:=

```
Clear[r, v, t, speed]

mag[vector_] := √(vector.vector)

Print["position vector is ",
 r[t_] = {2 t^.3, Cos[t - 1], Exp[-t^2]}]

Print["velocity vector is ", v[t_] = r'[t]]

Print["acceleration vector is ", a[t_] = v'[t

Print["speed is ", speed[t_] = mag[v[t]]]

Print["unit tangent vector is ",
 unittangent[t_] = v[t] / speed[t] // Simplify]
```

```
Print["curvature is ",
 curvature[t_] = mag[Cross[v[t], a[t]]] / Abs[:
   Simplify]

Print["torsion is ",
 torsion[t_] =
  Det[{v[t], a[t], a'[t]}] / mag[Cross[v[t], a

ParametricPlot3D[Evaluate[r[t]], {t, 0, 5},
 AxesLabel -> {x, y, z}];
```

position vector is $\left\{2\,t^{0.3}, \mathrm{Cos}[1-t], e^{-t^2}\right\}$

velocity vector is $\left\{\frac{0.6}{t^{0.7}}, \mathrm{Sin}[1-t], -2\,e^{-t^2}\,t\right.$

acceleration vector is
$\left\{-\frac{0.42}{t^{1.7}}, -\mathrm{Cos}[1-t], -2\,e^{-t^2}+4\,e^{-t^2}\,t^2\right\}$

speed is $\sqrt{\frac{0.36}{t^{1.4}}+4\,e^{-2t^2}\,t^2+\mathrm{Sin}[1-t]^2}$

unit tangent vector is

$$\left\{\frac{0.6}{t^{0.7}\sqrt{\frac{0.36}{t^{1.4}}+4\,e^{-2t^2}\,t^2+\mathrm{Sin}[1-t]^2}},\right.$$

$$\frac{\mathrm{Sin}[1-t]}{\sqrt{\frac{0.36}{t^{1.4}}+4\,e^{-2t^2}\,t^2+\mathrm{Sin}[1-t]^2}},$$

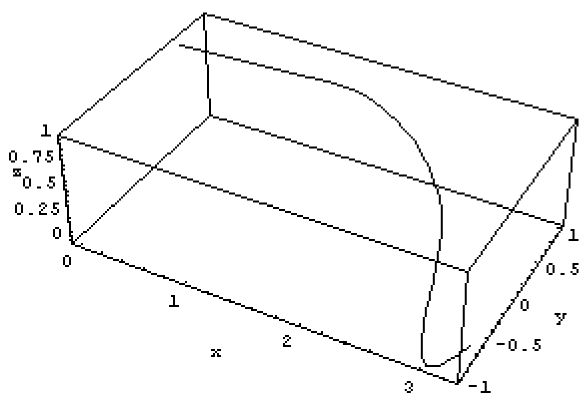
$$\left.-\frac{2\,e^{-t^2}\,t}{\sqrt{\frac{0.36}{t^{1.4}}+4\,e^{-2t^2}\,t^2+\mathrm{Sin}[1-t]^2}}\right\}$$

curvature is

$$\left(\sqrt{\left(\frac{e^{-2t^2}(2.04-2.4t^{2.})^2}{t^{1.4}}+\left(\frac{0.6\,\mathrm{Cos}[1-t]}{t^{0.7}}-\frac{0.42\,\mathrm{Sin}[1-t]}{t^{1.7}}\right)^2+4e^{-2t^2}(t\,\mathrm{Cos}[1-t]+(1-2t^2)\,\mathrm{Sin}[1-t])^2\right)}\right)\Big/\mathrm{Abs}\left[\frac{0.36}{t^{1.4}}+4e^{-2t^2}t^2+\mathrm{Sin}[1-t]^2\right]^{3/2}$$

torsion is

$$\left(e^{t^2}((-1.428t^{4.1}-7.2t^{6.1}+4.8t^{8.1})\,\mathrm{Cos}[1-t]\cdot(-1.428t^{3.1}+5.856t^{5.1}-0.96t^{7.1})\,\mathrm{Sin}[1-t])\right)\Big/\left(4.1616t^{4.4}-9.792t^{6.4}+5.76t^{8.4}+\left(0.36e^{2t^2}t^{4.4}+4.\,t^{7.8}\right)\mathrm{Cos}[1-t]^2+\left(-0.504e^{2t^2}t^{3.4}+8.\,t^{6.8}-16.\,t^{8.8}\right)\mathrm{Cos}[1-t]\,\mathrm{Sin}[1-t]+\left(0.1764e^{2t^2}t^{2.4}+4.\,t^{5.8}-16.\,t^{7.8}+16.\,t^{9.8}\right)\mathrm{Sin}[1-t]^2\right)$$



In[76]:=

```
speed[t]
```

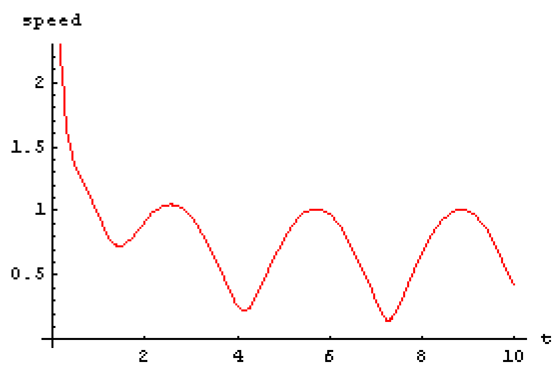
```
ps = Plot[Evaluate[speed[t]], {t, 0, 10}, Axes
  PlotStyle -> RGBColor[1, 0, 0]];
```

```
curvature[t]
```

```
pc = Plot[Evaluate[curvature[t]], {t, 0, 10},
  AxesLabel -> {t, curvature}, PlotRange -> All
  PlotStyle -> RGBColor[0, 1, 0]];

torsion[t]

pt = Plot[Evaluate[torsion[t]], {t, 0, 10},
  AxesLabel -> {t, torsion}, PlotStyle -> RGBC
  PlotRange -> All];

Show[ps, pc, pt];

Print["speed in red, curvature in green, tor
```

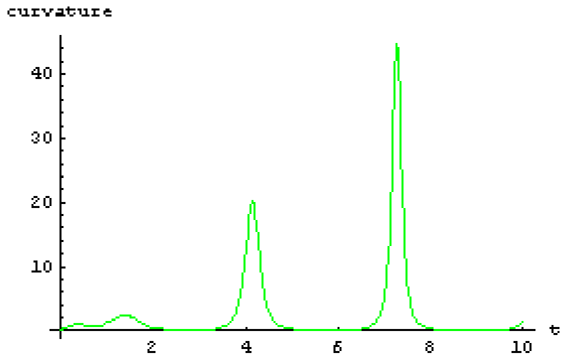
Out[76]=

$$\sqrt{\frac{0.36}{t^{1.4}} + 4 e^{-2 t^2} t^2 + \text{Sin}[1 - t]^2}$$



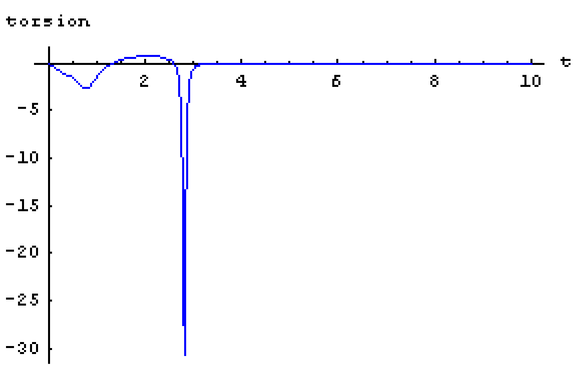
Out[78]=

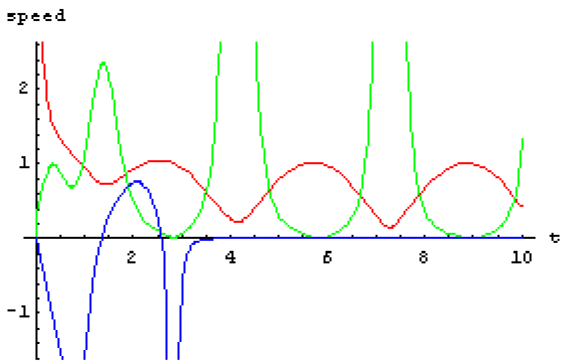
$$\left(\sqrt{\left(\frac{e^{-2 t^2} (2.04 - 2.4 t^{2.})^2}{t^{1.4}} + \left(\frac{0.6 \text{Cos}[1 - t]}{t^{0.7}} - \frac{0.42 \text{Sin}[1 - t]}{t^{1.7}}\right)^2 + 4 e^{-2 t^2} (t \text{Cos}[1 - t] + (1 - 2 t^2) \text{Sin}[1 - t])^2\right)}\right) \Big/ \text{Abs}\left[\frac{0.36}{t^{1.4}} + 4 e^{-2 t^2} t^2 + \text{Sin}[1 - t]^2\right]^{3/2}$$



Out[80]=

$$\Big(e^{t^2}\,((-1.428\,t^{4.1}-7.2\,t^{6.1}+4.8\,t^{8.1})\,\mathrm{Cos}[1-t]+(-1.428\,t^{3.1}+5.856\,t^{5.1}-0.96\,t^{7.1})\,\mathrm{Sin}[1-t])\Big)\Big/\Big(4.1616\,t^{4.4}-9.792\,t^{6.4}+5.76\,t^{8.4}+\big(0.36\,e^{2t^2}\,t^{4.4}+4.\,t^{7.8}\big)\,\mathrm{Cos}[1-t]^2+\big(-0.504\,e^{2t^2}\,t^{3.4}+8.\,t^{6.8}-16.\,t^{8.8}\big)\,\mathrm{Cos}[1-t]\,\mathrm{Sin}[1-t]+\big(0.1764\,e^{2t^2}\,t^{2.4}+4.\,t^{5.8}-16.\,t^{7.8}+16.\,t^{9.8}\big)\,\mathrm{Sin}[1-t]^2\Big)$$





speed in red, curvature
  in green, torsion in blue

Part V: TNB Frame-Computation and Visualization

The steps here resemble the ones above, except that we choose a simpler function and extend the formulas to the unit normal and binormal vectors.

In[84]:=

```
Clear[r, v, a, t, speed, curvature, torsion]

mag[vector_] := √vector.vector

Print["The position vector is ",
 r[t_] = {Sin[2 t], Cos[2 t], t^2/10}]

Print["The velocity vector is ", v[t_] = r'[t

Print["The acceleration vector is ", a[t_] =

Print["The speed is ", speed[t_] = mag[v[t]] ,

Print["The unit tangent vector is ",
 utan[t_] = v[t] / speed[t] // Simplify]

Print["The curvature is ",
 curvature[t_] = mag[Cross[v[t], a[t]]] / spee
```

```
Print["The torsion is ",
 torsion[t_] =
  Det[{v[t], a[t], a'[t]}] / mag[Cross[v[t], a

top = utan'[t] // Simplify;

bottom = mag[top] // Simplify;

Print["The unit normal is ", un[t_] = top / bo

Print["The unit binormal is ",
 ubn[t_] = Cross[utan[t], un[t]] // Simplify]
```

The position vector is

$\left\{\text{Sin}[2\,t],\ \text{Cos}[2\,t],\ \frac{t^2}{10}\right\}$

The velocity vector is

$\left\{2\,\text{Cos}[2\,t],\ -2\,\text{Sin}[2\,t],\ \frac{t}{5}\right\}$

The acceleration vector is

$\left\{-4\,\text{Sin}[2\,t],\ -4\,\text{Cos}[2\,t],\ \frac{1}{5}\right\}$

The speed is $\frac{\sqrt{100 + t^2}}{5}$

The unit tangent vector is

$\left\{\frac{10\,\text{Cos}[2\,t]}{\sqrt{100 + t^2}},\ -\frac{10\,\text{Sin}[2\,t]}{\sqrt{100 + t^2}},\ \frac{t}{\sqrt{100 + t^2}}\right\}$

The curvature is $\frac{50\sqrt{401 + 4\,t^2}}{(100 + t^2)^{3/2}}$

The torsion is $-\frac{40\,t}{401 + 4\,t^2}$

The unit normal is

$$\left\{-\frac{t\,\text{Cos}[2\,t]+2\,(100+t^2)\,\text{Sin}[2\,t]}{(100+t^2)^{3/2}\sqrt{\frac{401+4t^2}{(100+t^2)^2}}},\ \frac{-2\,(100+t^2)\,\text{Cos}[2\,t]+t\,\text{Sin}[2\,t]}{(100+t^2)^{3/2}\sqrt{\frac{401+4t^2}{(100+t^2)^2}}},\ \frac{10}{(100+t^2)^{3/2}\sqrt{\frac{401+4t^2}{(100+t^2)^2}}}\right\}$$

The unit binormal is

$$\left\{\frac{2\,t\,\text{Cos}[2\,t]-\text{Sin}[2\,t]}{(100+t^2)\sqrt{\frac{401+4t^2}{(100+t^2)^2}}},\ -\frac{\text{Cos}[2\,t]+2\,t\,\text{Sin}[2\,t]}{(100+t^2)\sqrt{\frac{401+4t^2}{(100+t^2)^2}}},\ -\frac{20}{(100+t^2)\sqrt{\frac{401+4t^2}{(100+t^2)^2}}}\right\}$$

Let's find the tangential and normal components of the accelerations.

In[97]:=

```
Print["The tangential component of accelerat
 at[t_] = a[t].utan[t] // Simplify]

Print["The normal component of acceleration
 an[t_] = a[t].un[t] // Simplify]
```

The tangential component of acceleration is $\frac{t}{5\sqrt{100+t^2}}$

The normal component of acceleration is

$$2\sqrt{100+t^2}\sqrt{\frac{401+4\,t^2}{(100+t^2)^2}}$$

You can get a visual perspective of the tangential and normal components of the acceleration in two dimensions by exploring the Java applet, "Tangent and Normal Vectors."

If you compare these components found by the dot products to the formulas given in your book, you will see that they agree, once the absolute value sign is removed from the curvature formula.

In[99]:=

```
at[t] == speed'[t] // Simplify

an[t] == curvature[t] speed[t]^2 // Simplify
```

Out[99]=

True

Out[100]=

$$2\sqrt{100+t^2}\sqrt{\frac{401+4t^2}{(100+t^2)^2}} == \frac{2\sqrt{401+4t^2}}{\sqrt{100+t^2}}$$

The equality of the tangential components is verified. Although *Mathematica* cannot completely simplify the results, you can see that the normal components are equal also.

The following set of commands will plot the unit tangent, unit normal, and unit binormal vectors as you move along the curve.

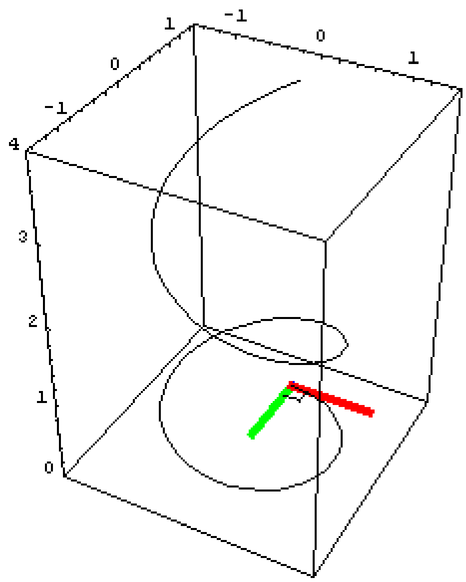
After the plots are generated, double click on any one to see the animation. Slow it down by clicking on the button in the bottom left corner of the notebook screen.

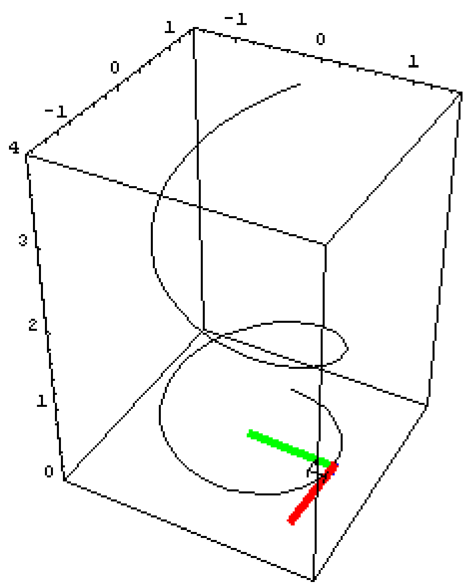
In[101]:=

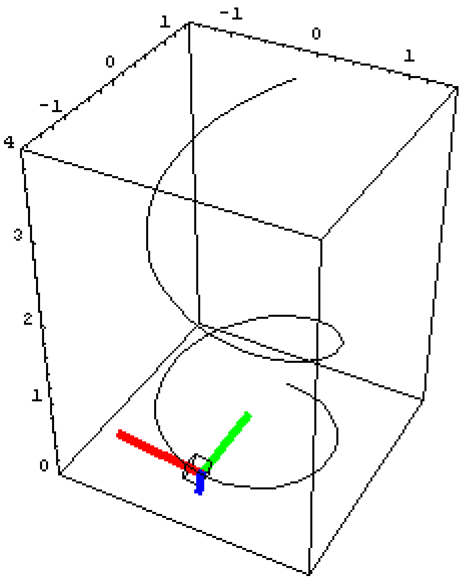
```
Off[ParametricPlot3D::"ppcom"];
```

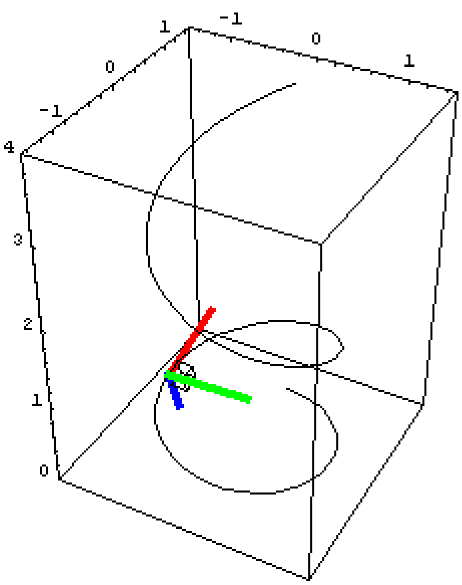
```
Do[
 p1 = ParametricPlot3D[r[t], {t, 0, 2. π},
   PlotRange → {{-1.5, 1.5}, {-1.5, 1.5}, {0, ·
   DisplayFunction → Identity];
p2 =
  Show[Graphics3D[{Thickness[0.02], RGBColor
     Line[{r[t], r[t] + utan[t]} /. t → ta]}],
   DisplayFunction → Identity];
 p3 =
  Show[Graphics3D[{Thickness[0.02], RGBColor
     Line[{r[t], r[t] + un[t]} /. t → ta]}],
   DisplayFunction → Identity];
p4 =
  Show[Graphics3D[{Thickness[0.02], RGBColor
     Line[{r[t], r[t] + ubn[t]} /. t → ta]}],
   DisplayFunction → Identity];
p5 =
  Show[Graphics3D[{Dashing[0.01, 0.01],
     Line[
      {r[t] + 0.20 * utan[t], r[t] + 0.20 * utan[
         0.20 * ubn[t]} /. t → ta],
     Line[
      {r[t] + 0.20 * ubn[t], r[t] + 0.20 * ubn[t]
       t → ta],
     Line[
      {r[t] + 0.20 * utan[t], r[t] + 0.20 * utan[
       t → ta],
     Line[
      {r[t] + 0.20 * un[t], r[t] + 0.20 * un[t] + (
       t → ta],
     Line[
      {r[t] + 0.20 * un[t], r[t] + 0.20 * un[t] + (
       t → ta],
     Line[
      {r[t] + 0.20 * ubn[t], r[t] + 0.20 * ubn[t]
       t → ta],
     Line[{r[t] + 0.20 * utan[t] + 0.20 * ubn[t],
        r[t] + 0.20 * utan[t] + 0.20 * ubn[t] + 0.
       t → ta],
     Line[{r[t] + 0.20 * utan[t] + 0.20 * un[t],
        r[t] + 0.20 * utan[t] + 0.20 * ubn[t] + 0.
       t → ta],
     Line[{r[t] + 0.20 * un[t] + 0.20 * ubn[t],
        r[t] + 0.20 * utan[t] + 0.20 * ubn[t] + 0.
       t → ta]}], DisplayFunction → Identity]
```

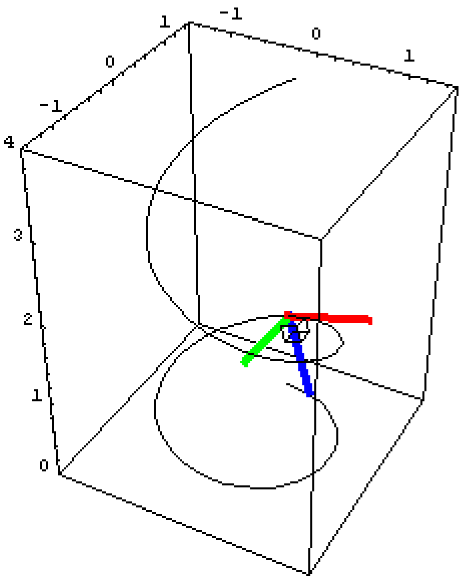
```
Print[
 "unit tangent in red, unit normal in green,
   binormal in blue"]
```

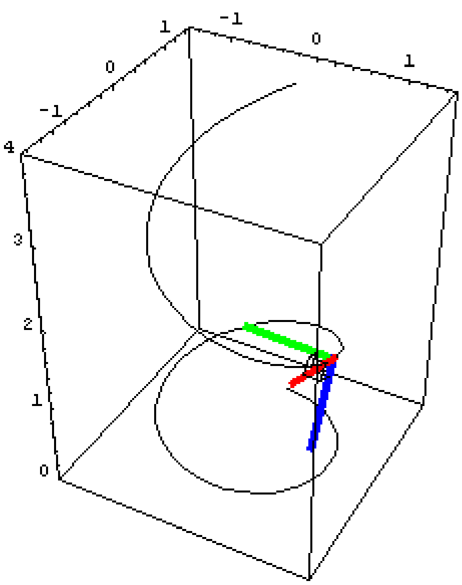


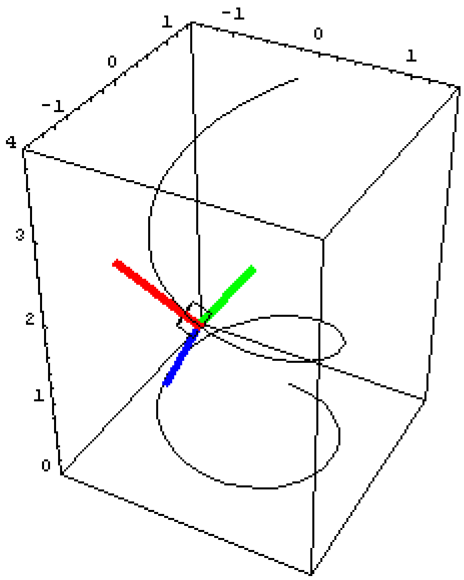


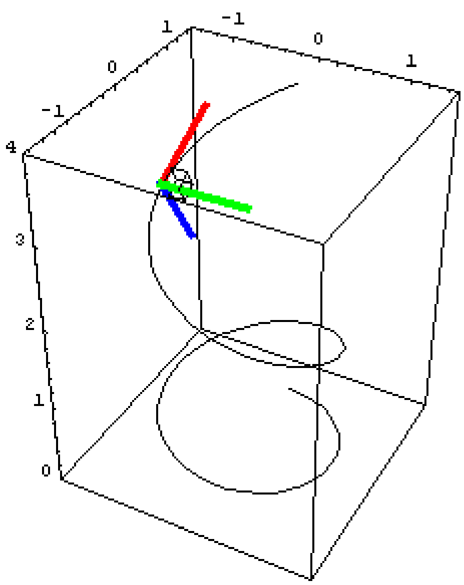

1
-1
0
1
0
-1
4
3
2
1
0

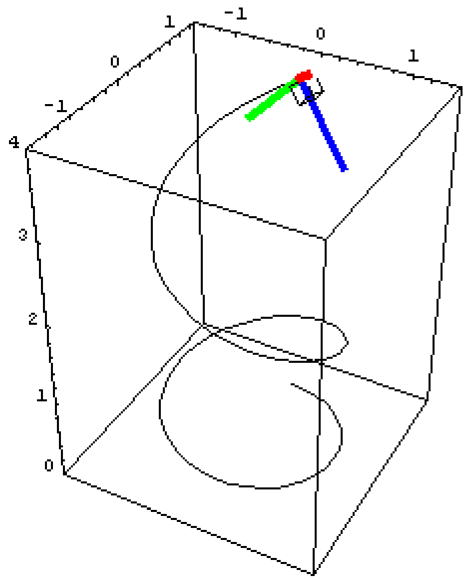

1
-1
0
1
0
-1
4
3
2
1
0


1
-1
0
1
0
-1
4
3
2
1
0


1
-1
0
1
0
-1
4
3
2
1
0


1
-1
0
1
0
-1
4
3
2
1
0


1
-1
0
1
0
-1
4
3
2
1
0




```
unit tangent in red, unit normal
  in green, unit binormal in blue
```

You Try It: Part V

The computational code is written so that all you have to do is change the initial vector function (in red) and then re-execute the commands. We suggest that you choose some of the homework problems from your text for the TNB frame, since the computation for the unit normal and unit binormal can get you very bogged down, even in *Mathematica*. You may also wish to change the domain over which you extend your function (red, in the second cell).

If you check out the following example, expect to wait a few minutes for the computations.

In[104]:=

```
Clear[r, v, t, speed]

mag[vector_] := √vector.vector

Print["position vector is ", r[t_] = {2 t, Cos

Print["velocity vector is ", v[t_] = r'[t]]

Print["acceleration vector is ", a[t_] = v'[t
```

```
Print["speed is ", speed[t_] = mag[v[t]]]

Print["unit tangent vector is ",
 utan[t_] = v[t] / speed[t] // Simplify]

Print["curvature is ",
 curvature[t_] = mag[Cross[v[t], a[t]]] / spee

Print["torsion is ",
 torsion[t_] =
  Det[{v[t], a[t], a'[t]}] / mag[Cross[v[t], a

top = utan'[t] // Simplify;

bottom = mag[top] // Simplify;

Print["unit normal is ", un[t_] = (top / botto

Print["unit binormal is ",
 ubn[t_] = Cross[utan[t], un[t]] // Simplify]
```

position vector is $\{2\,t, \mathrm{Cos}[1-t], e^{-t}\}$

velocity vector is $\{2, \mathrm{Sin}[1-t], -e^{-t}\}$

acceleration vector is $\{0, -\mathrm{Cos}[1-t], e^{-t}\}$

speed is $\sqrt{4+e^{-2t}+\mathrm{Sin}[1-t]^2}$

unit tangent vector is

$$\left\{\frac{2}{\sqrt{4+e^{-2t}+\mathrm{Sin}[1-t]^2}}, \frac{\mathrm{Sin}[1-t]}{\sqrt{4+e^{-2t}+\mathrm{Sin}[1-t]^2}}, -\frac{e^{-t}}{\sqrt{4+e^{-2t}+\mathrm{Sin}[1-t]^2}}\right\}$$

curvature is

$$\frac{\sqrt{2 + 5 e^{-2t} + 2 \text{Cos}[2-2t] - e^{-2t} \text{Sin}[2-2t]}}{(4 + e^{-2t} + \text{Sin}[1-t]^2)^{3/2}}$$

torsion is

$$\frac{2 e^{t} (\text{Cos}[1-t] + \text{Sin}[1-t])}{5 + 2 e^{2t} + 2 e^{2t} \text{Cos}[2-2t] - \text{Sin}[2-2t]}$$

unit normal is

$$\Big\{(2 e^{-2t} + \text{Sin}[2-2t]) \Big/ \Big(2 \sqrt{\big((e^{2t} (5 + 2 e^{2t} + 2 e^{2t} \text{Cos}[2-2t] - \text{Sin}[2-2t])) / (2 + 9 e^{2t} - e^{2t} \text{Cos}[2-2t])^2\big)} (4 + e^{-2t} + \text{Sin}[1-t]^2)^{3/2}\Big),$$

$$(-(1 + 4 e^{2t}) \text{Cos}[1-t] + \text{Sin}[1-t]) \Big/ \Big(2 \sqrt{\big((e^{2t} (5 + 2 e^{2t} + 2 e^{2t} \text{Cos}[2-2t] - \text{Sin}[2-2t])) / (2 + 9 e^{2t} - e^{2t} \text{Cos}[2-2t])^2\big)} \sqrt{4 + e^{-2t} + \text{Sin}[1-t]^2} (1 + 4 e^{2t} + e^{2t} \text{Sin}[1-t]^2)\Big),$$

$$-(e^{t} (-9 + \text{Cos}[2-2t] + \text{Sin}[2-2t])) \Big/ \Big(4 \sqrt{\big((e^{2t} (5 + 2 e^{2t} + 2 e^{2t} \text{Cos}[2-2t] - \text{Sin}[2-2t])) / (2 + 9 e^{2t} - e^{2t} \text{Cos}[2-2t])^2\big)} \sqrt{4 + e^{-2t} + \text{Sin}[1-t]^2} (1 + 4 e^{2t} + e^{2t} \text{Sin}[1-t]^2)\Big)\Big\}$$

unit binormal is

$$\Big\{\big(e^{t}\,(\text{Cos}[1-t]-\text{Sin}[1-t])\big)\Big/\Big((-2-9e^{2t}+e^{2t}\,\text{Cos}[2-2t])\sqrt{\Big(\big(e^{2t}\,(5+2e^{2t}+2e^{2t}\,\text{Cos}[2-2t]-\text{Sin}[2-2t])\big)\Big/(2+9e^{2t}-e^{2t}\,\text{Cos}[2-2t])^{2}\Big)}\Big),$$

$$(2e^{t})\Big/\Big((-2-9e^{2t}+e^{2t}\,\text{Cos}[2-2t])\sqrt{\Big(\big(e^{2t}\,(5+2e^{2t}+2e^{2t}\,\text{Cos}[2-2t]-\text{Sin}[2-2t])\big)\Big/(2+9e^{2t}-e^{2t}\,\text{Cos}[2-2t])^{2}\Big)}\Big),$$

$$(2e^{2t}\,\text{Cos}[1-t])\Big/\Big((-2-9e^{2t}+e^{2t}\,\text{Cos}[2-2t])\sqrt{\Big(\big(e^{2t}\,(5+2e^{2t}+2e^{2t}\,\text{Cos}[2-2t]-\text{Sin}[2-2t])\big)\Big/(2+9e^{2t}-e^{2t}\,\text{Cos}[2-2t])^{2}\Big)}\Big)\Big\}$$

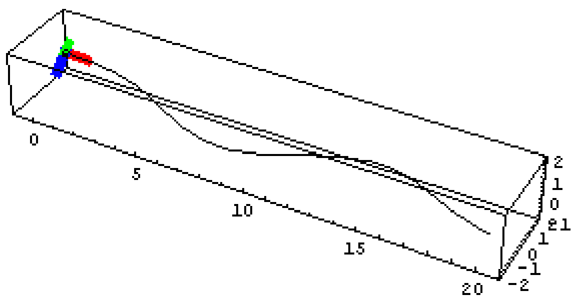
The following set of commands will plot the unit tangent, unit normal, and unit binormal vectors as you move along the curve.

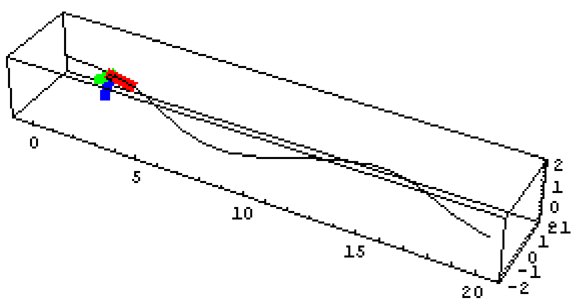
After the plots are generated, double click on any one to see the animation. Slow it down by clicking on the button in the bottom left corner of the notebook screen.

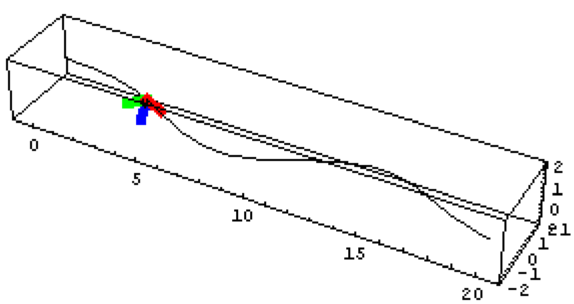
In[117]:=

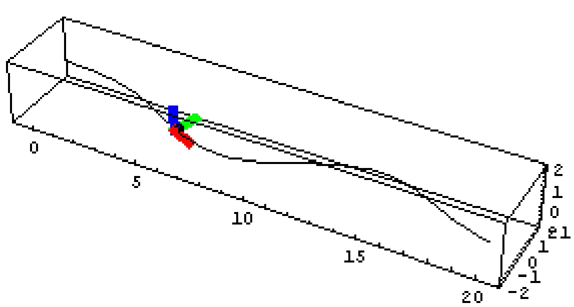
```
Off[ParametricPlot3D::"ppcom"];
Do[
 p1 = ParametricPlot3D[r[t], {t, 0, 10},
   PlotRange → {{-1, 21}, {-2, 2}, {-1, 2}},
   DisplayFunction → Identity];
p2 =
  Show[Graphics3D[{Thickness[0.02], RGBColor
     Line[{r[t], r[t] + utan[t]} /. t → ta]}],
   DisplayFunction → Identity];
 p3 =
  Show[Graphics3D[{Thickness[0.02], RGBColor
     Line[{r[t], r[t] + un[t]} /. t → ta]}],
   DisplayFunction → Identity];
p4 =
  Show[Graphics3D[{Thickness[0.02], RGBColor
     Line[{r[t], r[t] + ubn[t]} /. t → ta]}],
   DisplayFunction → Identity];
p5 =
  Show[Graphics3D[{Dashing[0.01, 0.01],
     Line[
      {r[t] + 0.20 * utan[t], r[t] + 0.20 * utan[
         0.20 * ubn[t]} /. t → ta],
     Line[
      {r[t] + 0.20 * ubn[t], r[t] + 0.20 * ubn[t]
       t → ta],
     Line[
      {r[t] + 0.20 * utan[t], r[t] + 0.20 * utan[
       t → ta],
     Line[
      {r[t] + 0.20 * un[t], r[t] + 0.20 * un[t] + (
       t → ta],
     Line[
      {r[t] + 0.20 * un[t], r[t] + 0.20 * un[t] + (
       t → ta],
     Line[
      {r[t] + 0.20 * ubn[t], r[t] + 0.20 * ubn[t]
       t → ta],
     Line[{r[t] + 0.20 * utan[t] + 0.20 * ubn[t],
        r[t] + 0.20 * utan[t] + 0.20 * ubn[t] + 0.
       t → ta],
     Line[{r[t] + 0.20 * utan[t] + 0.20 * un[t],
        r[t] + 0.20 * utan[t] + 0.20 * ubn[t] + 0.
       t → ta],
     Line[{r[t] + 0.20 * un[t] + 0.20 * ubn[t],
        r[t] + 0.20 * utan[t] + 0.20 * ubn[t] + 0.
```

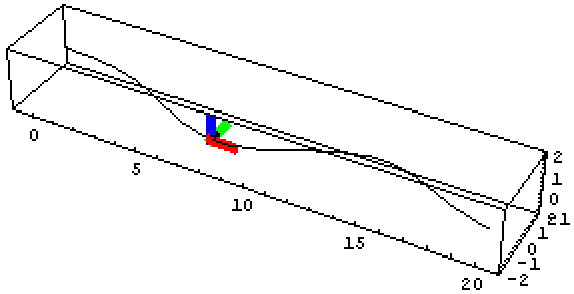
```
Print[
 "unit tangent in red, unit normal in green,
   binormal in blue"]
```

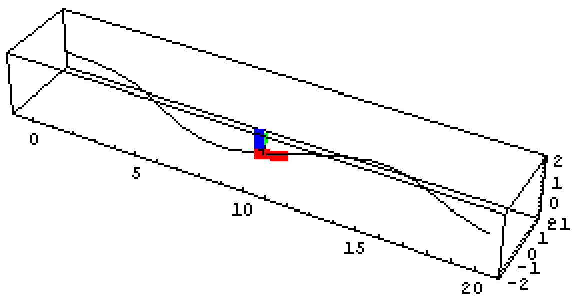


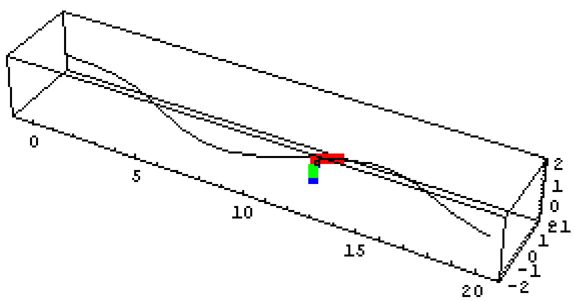


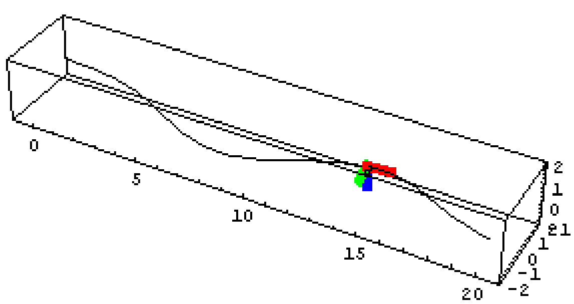


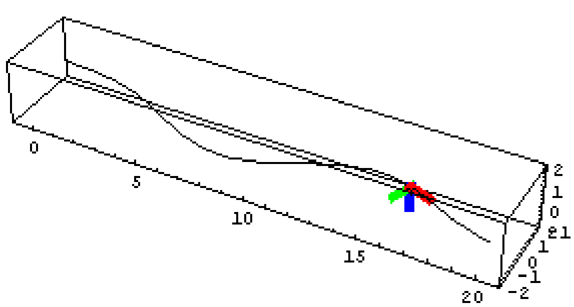


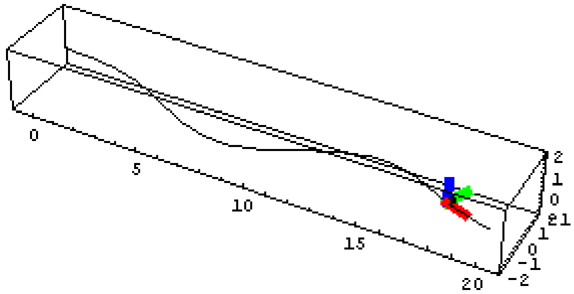

0
5
10
15
20
2
1
0
-1
-2

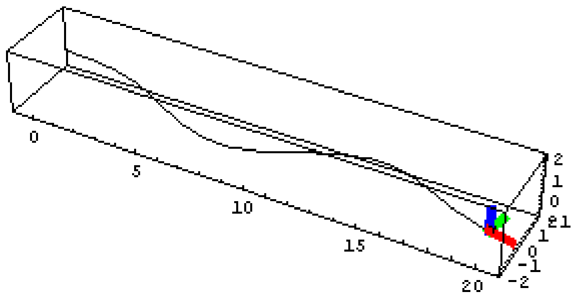

0
5
10
15
20
2
1
0
-1
-2


0
5
10
15
20
2
1
0
-1
-2


0
5
10
15
20
2
1
0
-1
-2


0
5
10
15
20
2
1
0
-1
-2





unit tangent in red, unit normal
in green, unit binormal in blue

Try some more problems from your text.

About *Mathematica*

In *Mathematica*, the derivative can be found with either the derivative command **D [function,variable to differentiate with respect to]** or the prime (single quote), **f'[x]**. For the prime to work, the function must be defined with an argument (**f[x]**), whereas the differentiation command can be applied to a function in any form, with or without an argument.
Go back

Created by *Mathematica* (April 2, 2005)