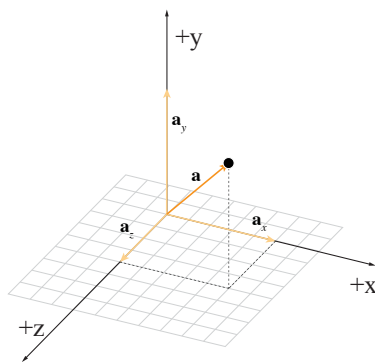


Coordinates

1. [Cartesian Coordinates](#)
2. [Vector Math](#)
3. [Other Coordinate systems](#)
 1. [Polar](#)
 2. [Example](#)

1. Cartesian Coordinates



The Cartesian Coordinate system

The cartesian 3d coordinate system

There are several common ways to express a location:

$$\mathbf{a} = a_x \hat{\mathbf{x}} + a_y \hat{\mathbf{y}} + a_z \hat{\mathbf{z}} \quad (1)$$

$$= a_x \hat{\mathbf{i}} + a_y \hat{\mathbf{j}} + a_z \hat{\mathbf{k}} \quad (2)$$

$$= (a_x, a_y, a_z) \quad (3)$$

$$= a_1 \mathbf{e}_1 + a_2 \mathbf{e}_2 + a_3 \mathbf{e}_3 \quad (4)$$

$$= \sum_{i=1}^3 a_i \mathbf{e}_i \quad (5)$$

$$= \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (6)$$

In a 3d world, we can describe the position of any point using 3 coordinates.

This of course is a **vector**!

2. Vector Math

Magnitudes

$$a^2 + b^2 = c^2$$

Addition & Subtraction

$$\mathbf{a} = a_1 \hat{\mathbf{x}} + a_2 \hat{\mathbf{y}} + a_3 \hat{\mathbf{z}}$$

And

$$\mathbf{b} = b_1 \hat{\mathbf{x}} + b_2 \hat{\mathbf{y}} + b_3 \hat{\mathbf{z}}$$

then,

$$\mathbf{a} + \mathbf{b} = (a_1 + b_1) \hat{\mathbf{x}} + (a_2 + b_2) \hat{\mathbf{y}} + (a_3 + b_3) \hat{\mathbf{z}}$$

and

$$\mathbf{a} - \mathbf{b} = (a_1 - b_1) \hat{\mathbf{x}} + (a_2 - b_2) \hat{\mathbf{y}} + (a_3 - b_3) \hat{\mathbf{z}}$$

Dot Product

There are two ways of multiplying vectors:

1. The dot product (or scalar product)

$$\mathbf{a} \cdot \mathbf{b} = ab \cos \theta$$

$$\mathbf{a} \cdot \mathbf{b} = (a_1 \hat{\mathbf{x}} + a_2 \hat{\mathbf{y}} + a_3 \hat{\mathbf{z}}) \cdot (b_1 \hat{\mathbf{x}} + b_2 \hat{\mathbf{y}} + b_3 \hat{\mathbf{z}})$$

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + a_3 b_3$$

Dot Product - Linear Algebra

$$\begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} \cdot \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = a_1 b_1 + a_2 b_2 + a_3 b_3$$

Cross Product

Defined as:

$$\mathbf{a} \times \mathbf{b} = |\mathbf{a}| |\mathbf{b}| \sin(\theta) \mathbf{n}$$

where θ is the angle between \mathbf{a} and \mathbf{b} , and \mathbf{n} is the unit vector perpendicular to the plane containing \mathbf{a} and \mathbf{b}

This produces a third vector that points perpendicular to both the original vectors.

$$\mathbf{a} \times \mathbf{b} = (a_x \hat{\mathbf{x}} + a_y \hat{\mathbf{y}} + a_z \hat{\mathbf{z}}) \times (b_x \hat{\mathbf{x}} + b_y \hat{\mathbf{y}} + b_z \hat{\mathbf{z}})$$

We can calculate the cross product by computing the determinant of this matrix:

$$\mathbf{a} \times \mathbf{b} = \begin{vmatrix} \hat{\mathbf{x}} & \hat{\mathbf{y}} & \hat{\mathbf{z}} \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix}$$

$$\mathbf{a} \times \mathbf{b} = (a_y b_z - a_z b_y) \hat{\mathbf{x}} + (a_z b_x - a_x b_z) \hat{\mathbf{y}} + (a_x b_y - a_y b_x) \hat{\mathbf{z}}$$

Mathematica:

```
a = {a1, a2, a3}
b = {b1, b2, b3}
Dot[a,b]
-> a1 b1 + a2 b2 + a3 b3
Cross[a,b]
-> {-a3 b2 + a2 b3, a3 b1 - a1 b3, -a2 b1 + a1 b2}
```

numpy (python)

```
import numpy as np

a = np.array([2,4,5])
b = np.array([1,-2,3])
c = np.dot(a,b)
print(c)
-> 9
d = np.cross(a,b)
print(d)
-> [22 -1 -8]
```

Quadratic Drag in two dimensions

In cartesian coordinates:

$$m\ddot{\mathbf{r}} = m\mathbf{g} - cv^2 \hat{\mathbf{v}} \quad (7)$$

$$= m\mathbf{g} - cv \mathbf{v} \quad (8)$$

This, if you remember 2 kinematics from 20700, requires breaking the problem up into both directions. Previously, those two directions were independent, meaning what happens in one, stays there and doesn't affect the other. But, know, with the velocity dependent drag, they are coupled:

$$m\dot{v}_x = -c\sqrt{v_x^2 + v_y^2} v_x \quad (9)$$

$$m\dot{v}_y = -mg - c\sqrt{v_x^2 + v_y^2} v_y \quad (10)$$

This is an example (already!) of something that has no analytic solution. That means we can't use separation of variable or any other regular diff.eq techniques to find a general solution to this. Numerical methods are the only way.

Example Problem #1:

Plot motion for a projectile experiencing quadratic drag.

Mathematica:

Define some parameters

```
r = .05;  
area = Pi r^2;  
m = .15;  
g = 9.8;  
Cd = 0.47;  
\[Rho] = 1.29;  
c = 1/2 \[Rho] area Cd;  
\[Theta] = Pi/4  
v0 = 40;
```

Define our differential equations

In the x direction:

$$m\ddot{x} = -c\dot{x}\sqrt{\dot{x}^2 + \dot{y}^2}$$

And

$$m\ddot{y} = -mg - c\dot{y}\sqrt{\dot{x}^2 + \dot{y}^2}$$

```

diffeqs = {m x''[t] == -c x'[t]*Sqrt[x'[t]^2 + y'[t]^2],
  m y''[t] == -m g - c y'[t]*Sqrt[x'[t]^2 + y'[t]^2],
  x[0] == 0,
  x'[0] == v0 Cos[\[Theta]],
  y[0] == 1,
  y'[0] == v0 Sin[\[Theta]]
}

```

Use `NDSolve[]` to solve them.

```

solutions = NDSolve[diffeqs, {x[t], y[t]}, {t, 0, 6}]

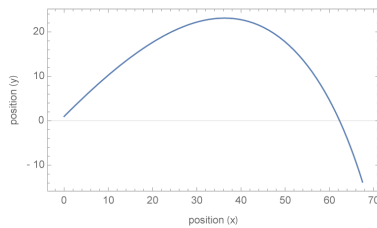
```

Plot x vs. y using `ParametricPlot[]`

```

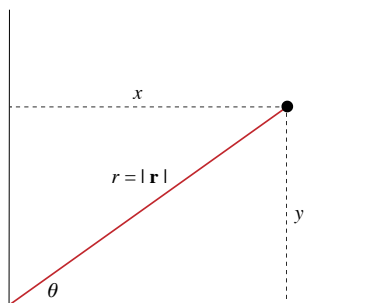
ParametricPlot[{x[t] /. solutions[[1, 1]], y[t] /. solutions[[1, 2]]}, {t, 0, 5}]

```



3. Other Coordinate systems

3.1 Polar



The basic polar coordinate system

$$x = r \cos \theta \quad (11)$$

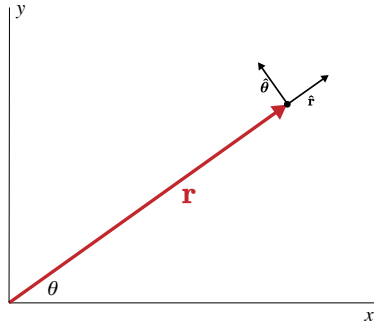
$$y = r \sin \theta$$

$$r = \sqrt{x^2 + y^2} \quad (12)$$

$$\theta = \tan^{-1} \left(\frac{y}{x} \right)$$

Goal: Express newton's laws in these non-cartesian coordinates:

$$\mathbf{F} = m\ddot{\mathbf{r}}$$

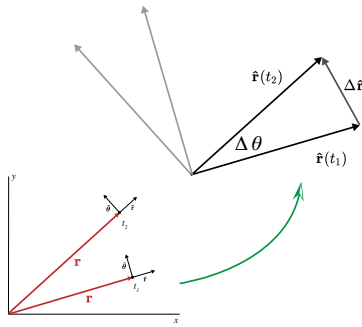


Unit vectors in polar coordinates

$\hat{\mathbf{r}}$ points in the direction of increasing r (given a fixed θ)

$\hat{\boldsymbol{\theta}}$ points in the direction of increasing θ (given a fixed r)

Unit vectors in polar coordinates



Now, the unit vectors change!

We start with $\hat{\mathbf{r}}$

$$\Delta \hat{\mathbf{r}} \approx \Delta \theta \hat{\boldsymbol{\theta}} \quad (13)$$

$$\approx \dot{\theta} \Delta t \hat{\boldsymbol{\theta}} \quad (14)$$

Taking the time derivative:

$$\frac{d\hat{\mathbf{r}}}{dt} = \dot{\theta} \hat{\boldsymbol{\theta}}$$

Taking the time derivative of $\mathbf{r} = r\hat{\mathbf{r}}$

$$\dot{\mathbf{r}} = \dot{r}\hat{\mathbf{r}} + r\frac{d\hat{\mathbf{r}}}{dt} \quad (15)$$

which can be used to express the velocity of the particle in polar coordinates:

$$\mathbf{v} \equiv \dot{\mathbf{r}} = \dot{r}\hat{\mathbf{r}} + r\dot{\theta}\hat{\boldsymbol{\theta}} \quad (16)$$

This might look a little different, but you've seen it before. The components are:

$$v_r = \dot{r} \quad (17)$$

$$v_\theta = r\dot{\theta} = r\omega \quad (18)$$

(18) is just what we had for **Uniform Circular Motion**, i.e. when r wasn't changing. Now we have to account for changing r values, so we really need both terms.

Velocity expressed in polar coordinates:

$$\dot{\mathbf{r}} = \dot{r}\hat{\mathbf{r}} + r\dot{\theta}\hat{\boldsymbol{\theta}} \quad (19)$$

Now we need to get the acceleration

To find the acceleration, we now have to take the time derivative of the velocity:

$$\mathbf{a} \equiv \ddot{\mathbf{r}} = \frac{d}{dt}\dot{\mathbf{r}} = \frac{d}{dt}(\dot{r}\hat{\mathbf{r}} + r\dot{\theta}\hat{\boldsymbol{\theta}}) \quad (20)$$

The new term to deal with is

$$\frac{d\hat{\boldsymbol{\theta}}}{dt}$$

which by similar arguments as above, we can see will be:

$$\frac{d\hat{\theta}}{dt} = -\dot{\theta}\hat{r} \quad (21)$$

Add this back into our acceleration equation:

$$\mathbf{a} = \left(\ddot{r}\hat{r} + \dot{r}\frac{d\hat{r}}{dt} \right) + \left((\dot{r}\dot{\theta} + r\ddot{\theta})\hat{\theta} + r\dot{\theta}\frac{d\hat{\theta}}{dt} \right) \quad (22)$$

Acceleration in polar coordinates:

$$\mathbf{a} = (\ddot{r} - r\dot{\theta}^2)\hat{r} + (r\ddot{\theta} + 2\dot{r}\dot{\theta})\hat{\theta} \quad (23)$$

Example Problem
#2:

3.2 Example

A mass m is spun around in a circle of radius R at angular velocity ω . Using Newton's 2nd law in polar coordinates, find the tension in the string. No other forces are present.

Starting with

$$\mathbf{a} = (\ddot{r} - r\dot{\theta}^2)\hat{r} + (r\ddot{\theta} + 2\dot{r}\dot{\theta})\hat{\theta}$$

We can express the two Force components as:

$$F_r = m(\ddot{r} - r\dot{\theta}^2) \quad (24)$$

and

$$F_\theta = m(r\ddot{\theta} + 2\dot{r}\dot{\theta}) \quad (25)$$

Things simplify since $\dot{r} = \ddot{r} = 0$, and $\ddot{\theta} = 0$

The only force acting is in the (-) radial direction, thus:

$$F_r = -T = -mR\omega^2 \quad (26)$$

This is the familiar centripetal force, mv^2/R since $\omega = v/R$