GEOMETRY AND VECTORS
Distinguishing Between Points in Space

- **One Approach** – **Names**: ("Fred", "Steve", "Alice"...)
  - **Problem**: distance & direction must be defined point-by-point

- **More elegant** – take advantage of **geometry**
  - Label points in organized fashion with numbers ("coordinates")
  - Use the coordinates to **calculate** distance & direction
  - **Example**: Instead of "Chicago" → “42º N, 88º W”

- **How to choose coordinates for each point?**
  - **Common approach**: pick a reference point (the “origin”)
  - Label each point P with:
    - distance(origin, P) and direction(origin, P)
Measuring Direction – Projection

To make a “coordinate system” for a given space:

- Must quantitatively define $\text{direction}(A, B)$

Approach: Pick some direction to act as reference

- Quantitatively compare $\text{direction}(A, B)$ to reference direction
- This comparison is called a “projection”
- Measures how much of distance $\text{distance}(A, B)$ is parallel to reference
- Expressed as an angle or a number (between -1.0 and 1.0)

Reference direction (also called a coordinate axis)

In trigonometry, projection is represented by the cosine of the angle between $\text{direction}(A, B)$ and the reference direction
Dimensions

- Points in a space can be organized a variety of ways
  - Depending on how they are “connected” to each other

- **Dimension** of a space
  - Smallest # of coordinates necessary to specify each point
  - In defining “direction” → 1 “reference direction” per dimension

<table>
<thead>
<tr>
<th>1-Dimension</th>
<th>2-Dimensions</th>
<th>3-Dimensions</th>
<th>N-Dimensional</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
</tbody>
</table>
Reference Frames

- Ref. Frame – specific origin and reference directions
  - “Conventions” – man-made rules; convenient but not mathematically necessary:
  - Reference directions (called “coordinate axes”) are orthogonal
  - 3-Dimensions: coordinate axes xyz named by “right-hand rule”

- Two conventions for “naming” points in D dimensions:
  - 1) project Dist(O, P) onto D coordinate axes (Example: x, y, z)
  - 2) Dist(O,P) and projection onto D-1 axes (Example: r, θ, φ)
# Common Coordinate System Conventions

## 1-Dimension

<table>
<thead>
<tr>
<th>Coordinate: $x$</th>
</tr>
</thead>
</table>

## 2-Dimensions

| “Cartesian” Coordinates: $x, y$ |
| “Polar” Coordinates: $r, \theta$ |

## 3-Dimensions

| “Cartesian” Coordinates: $(x, y, z)$ |
| “Cylindrical” Coordinates: $(\rho, \phi, z)$ or $(r, \theta, z)$ |
| “Spherical” Coordinates: $(r, \theta, \phi)$ or $(\rho, \theta, \phi)$ |

---

**Note:** Math and Physics use different conventions for spherical coordinates.
## Coordinate System Consistency

- Geometry → can “translate” coordinates between systems

### 2-Dimensions

<table>
<thead>
<tr>
<th>Cartesian / Polar</th>
<th>Cartesian / Cylindrical</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r = \sqrt{x^2 + y^2} )</td>
<td>( \rho = \sqrt{x^2 + y^2} )</td>
</tr>
<tr>
<td>( \theta = \tan^{-1}\left(\frac{y}{x}\right) )</td>
<td>( \phi = \tan^{-1}\left(\frac{y}{x}\right) )</td>
</tr>
<tr>
<td>( x = r \cos(\theta) )</td>
<td>( x = \rho \cos(\phi) )</td>
</tr>
<tr>
<td>( y = r \sin(\theta) )</td>
<td>( y = \rho \sin(\phi) )</td>
</tr>
<tr>
<td>( z = z )</td>
<td>( z = z )</td>
</tr>
</tbody>
</table>

(\textit{actually arctan}2)

### 3-Dimensions

<table>
<thead>
<tr>
<th>Cartesian / Spherical</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r = \sqrt{x^2 + y^2 + z^2} )</td>
</tr>
<tr>
<td>( \theta = \cos^{-1}\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right) )</td>
</tr>
<tr>
<td>( \phi = \tan^{-1}\left(\frac{y}{x}\right) )</td>
</tr>
<tr>
<td>( r = r \sin(\theta) \cos(\phi) )</td>
</tr>
<tr>
<td>( y = r \sin(\theta) \sin(\phi) )</td>
</tr>
<tr>
<td>( z = r \cos(\theta) )</td>
</tr>
</tbody>
</table>
Spatial “Transformations”

• Space itself is isotropic → symmetric in all directions
  - There is no universal up, down, left, right – it's all convention
  - No point in space is “special” or distinguishable from others

• So any choice for origin and coordinate axes is valid
  - Physics needs to work for all reference frames!
  - Tricky: In different reference frames...
  - ...same point has different coordinates!

• How can reference frames differ?
  - Translation – different origins
  - Rotation – different coordinate axes
  - Scaling – coordinates are multiplied by a constant
Vector Spaces

- Can scale reference frame using any constant quantity
  - Even one with units! (kg, m, sec, or any multiplicative mix)
  - Creates a new “space” with same directions but different units

- **Vector space** – mathematical generalization of “space”
  - Which may or may not represent actual physical space
  - Generalized term for points in a vector space: “Vectors”
  - Generalized term for coordinates: “Components”
  - Generalized term for Distance\((O, P')\): “Magnitude”
Vector Spaces – Conventions

- **Vector symbol:** letter with arrow (\( \vec{A} \)) or boldface (\( A \))
  - Drawn graphically as an arrow directed from tail to tip
  - Magnitude is denoted by absolute value (\( |A| \)) or letter only (A)

- Can use usual coordinate systems (cartesian, polar...):
  - “Magnitude form” of a vector:
    - Magnitude (in any units) and direction – usually angle(s)
    - Example: \( a = 40.3 \text{ m/s}^2 \) and \( \theta = 73.2^\circ \)
  - “Component form” of a vector:
    - One component for each dimension
    - Example: \( (v_x = 3.0 \text{ m/s}, \ v_y = 4.1 \text{ m/s}, \ v_z = 2.2 \text{ m/s}) \)
“Adding” Vectors

• In physical space:
  - Every **point** is associated with a **position vector**
  - 2 **different** points are connected by a **displacement vector**
  - Conventional notation: \[ \vec{r}_B = \vec{r}_A + \vec{d}_{A\rightarrow B} \]

• “+” operation can be generalized to any vector space

• For vectors in **component form**: \[ \vec{A} = \vec{B} + \vec{C} \]
  - **Note:** Impossible to add vectors from **different** vector spaces
  - (components would have different **units** → makes no sense)
  - **Ex:** Cannot add a **displacement** vector to a **velocity** vector!
Unit Vectors

- Vectors which are used only to define direction
  - **Magnitude**: dimensionless and equal to 1

- **Convention**: Unit vectors in the x, y, z directions
  - Are called \( \hat{i}, \hat{j}, \hat{k} \) or \( \hat{x}, \hat{y}, \hat{z} \)

- Can construct a unit vector in any direction
  - With combinations of \( \hat{i}, \hat{j}, \hat{k} \)

**Common vector notations:**

\[
\vec{v} = v_x \hat{i} + v_y \hat{j} + v_z \hat{k}
\]

\[
\vec{v} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}
\]

For any vector \( \vec{v} \):

\[
\hat{v} \equiv \frac{\vec{v}}{v}
\]

\[
\hat{n} = \left( \frac{1}{\sqrt{2}} \right) \hat{i} + \left( \frac{1}{\sqrt{2}} \right) \hat{j} + 0 \hat{k}
\]
Actual Physical Space – Conventions

- **Displacement Vector**
  - Vector from any point A to any point B

- **Position Vector** (denoted by $\mathbf{r}$ or $\mathbf{x}$)
  - From origin to any point P → Components: $x$, $y$, $z$

\[
\mathbf{r}_A = x_A \hat{i} + y_A \hat{j} + z_A \hat{k}
\]
\[
\mathbf{d} = (x_B - x_A) \hat{i} + (y_B - y_A) \hat{j} + (z_B - z_A) \hat{k}
\]

Magnitudes of Vectors in “position space”:

Measured in units of length

\[
|\mathbf{r}_A| \equiv r_A = \sqrt{x_A^2 + y_A^2 + z_A^2}
\]
\[
|\mathbf{d}| \equiv d = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2}
\]
Coordinate Transformations

• To describe the same point in 2 reference frames:
  – Need to “transform” coordinates between frames

Translating a reference frame

\[ \vec{r}_A' = \vec{r}_A - \vec{R} \]

\[
\begin{pmatrix}
  x_A' \\
  y_A'
\end{pmatrix} = \begin{pmatrix}
  x_A - R_x \\
  y_A - R_y
\end{pmatrix}
\]

Rotating a reference frame about the z-axis

\[
\begin{pmatrix}
  x_A' \\
  y_A'
\end{pmatrix} = \begin{pmatrix}
  x_A \cos \phi + y_A \sin \phi \\
  -x_A \sin \phi + y_A \cos \phi
\end{pmatrix}
\]

To rotate about an axis other than z:
Similar concept with more complicated geometry
Rotation Matrix

- **Matrix** – structure for organizing numbers or functions
  - Matrices can “operate” on a vector (making a new vector)
  - Operations → carried out in specific order (rows and columns)

\[
\begin{pmatrix}
a_1 \\
a_2
\end{pmatrix} =
\begin{pmatrix}
M_{11} & M_{12} \\
M_{21} & M_{22}
\end{pmatrix}
\begin{pmatrix}
v_1 \\
v_2
\end{pmatrix}
\equiv
\begin{pmatrix}
M_{11} v_1 + M_{12} v_2 \\
M_{21} v_1 + M_{22} v_2
\end{pmatrix}
\]

In index notation:
\[a_i \equiv M_{ij} v_j\]

- **Rotation Matrix** – defines coordinate transformation
  - To a frame rotated about a particular axis by angle \( \Phi \)
  - For rotation about the z-axis:

\[
R(\Phi) =
\begin{pmatrix}
\cos \Phi & \sin \Phi & 0 \\
-\sin \Phi & \cos \Phi & 0 \\
0 & 0 & 1
\end{pmatrix}
\]

\[
\vec{r}' = R(\Phi) \vec{r}
\]
Vectors and Unit Vectors – Examples

• Let $\vec{A} = (2\,m)\hat{i} + (3\,m)\hat{j} - (1\,m)\hat{k}$ and $\vec{B} = (5\,m)\hat{i} - (2\,m)\hat{j} - (3\,m)\hat{k}$
  – In some particular reference frame S

• Consider a new reference frame S'
  – With the same origin as S, but rotated 45° about the z-axis

• In both reference frames:
  – Calculate the components of $\hat{\vec{A}}$ and $\hat{\vec{B}}$
  – Calculate $|\vec{A} + \vec{B}|$ and $|\vec{A} - \vec{B}|$
Unit Vectors in Polar Coordinates

- Vector components in Cartesian coordinates:
  - Are projections onto fixed directions xyz

- An alternate method for defining components:
  - Use projections parallel and perpendicular to position vector
  - Unit vectors in these directions are called $\hat{r}$ and $\hat{\theta}$
  - Note: These unit vectors depend on position (not fixed!)

$$\vec{A} = A_r \hat{r} + A_\theta \hat{\theta}$$

$$\vec{r} = r \hat{r}$$

$$\hat{r} = \cos(\theta) \hat{i} + \sin(\theta) \hat{j}$$

$$\hat{\theta} = -\sin(\theta) \hat{i} + \cos(\theta) \hat{j}$$
Graphical Representation of Vectors

- Vectors → defined by **direction** and **magnitude** only
  - Their “**location**” in the vector space is arbitrary

- Can move vectors around to use geometry
  - With the role of **distance** replaced by vector **magnitudes**

---

Comparing the directions of 2 vectors (i.e. measuring angle between them)

- **“Tail-to-tail”** convention:
  \[ \vec{A} + \vec{B} = \vec{C} \]

  **Note**: It is possible to compare directions of 2 vectors in **different** vector spaces

- **“Tail-to-tip”** convention:
  \[ \theta_{AB} + \theta_{BC} + \theta_{AC} = 180 \]

Geometry: These 3 vectors form a triangle in their vector space
Dot Product

- Angle measurement compares **direction** of 2 vectors
  - Can be tricky to do with vectors in component form

- **Useful tool**: the “dot product”
  - Measures how one vector **projects** onto another
  - Can be defined in either magnitude form or component form

\[
\vec{A} \cdot \vec{B} = |\vec{A}| |\vec{B}| \cos(\theta_{AB})
\]

\[
\vec{A} \cdot \vec{B} = A_x B_x + A_y B_y + A_z B_z
\]

- Dot product can be positive, negative, or zero
- Units of dot product: multiply units of individual vectors
- Also called “**scalar product**” or “**inner product**”
Dot Product – Important Features

• Dot product is “invariant”
  – Has the same value in all reference frames
    \[ \vec{A} \cdot \vec{B} = A_x B_x + A_y B_y + A_z B_z \]
  – \( A_x, B_x, A_y, \) etc. depend on frame but dot product does not

• Dot product is commutative: \( \vec{U} \cdot \vec{V} = \vec{V} \cdot \vec{U} \)

• Can take dot product of a vector with itself (\( \vec{A} \cdot \vec{A} \))
  – Result: “magnitude squared” of the vector (\( A^2 \))

• Dot products of unit vectors:
  \[ \hat{i} \cdot \hat{i} = \hat{j} \cdot \hat{j} = \hat{k} \cdot \hat{k} = 1 \quad \hat{i} \cdot \hat{j} = \hat{j} \cdot \hat{k} = \hat{k} \cdot \hat{i} = 0 \]
Cross Product

• Any 2 vector directions define a **plane**
  - Ways to mathematically describe the plane:
    1) “Equation of constraint” governing coordinates of points
    2) **Direction** which is perpendicular to the plane

• **“Cross Product”** of 2 vectors \( \vec{A} \) and \( \vec{B} \)
  - Produces a 3\(^{rd} \) vector \( \vec{C} \) with the properties:
    - **Direction**: perpendicular to both \( \vec{A} \) and \( \vec{B} \)
    - **Magnitude**: “area enclosed” by \( \vec{A} \) and \( \vec{B} \)

\[
|\vec{A} \times \vec{B}| = |\vec{A}| |\vec{B}| \sin(\theta_{AB})
\]

\[
\vec{A} \times \vec{B} = (A_y B_z - A_z B_y) \hat{i} - (A_x B_z - A_z B_x) \hat{j} + (A_x B_y - A_y B_x) \hat{k}
\]

Convention: Direction of cross product decided by “right-hand rule”
Cross Product – Important Features

• Cross product is a vector → frame-dependent
  – Components depend on reference frame – magnitude doesn't

• Cross product is anti-commutative: \( \vec{A} \times \vec{B} = - \vec{B} \times \vec{A} \)

• Cross product of a vector with itself (\( \vec{A} \times \vec{A} \)) is zero

• Cross products of unit vectors:
  
  \[
  \begin{align*}
  \hat{i} \times \hat{j} &= \hat{k} \\
  \hat{j} \times \hat{k} &= \hat{i} \\
  \hat{k} \times \hat{i} &= \hat{j}
  \end{align*}
  \]

  \[
  \begin{align*}
  \hat{j} \times \hat{i} &= -\hat{k} \\
  \hat{k} \times \hat{j} &= -\hat{i} \\
  \hat{i} \times \hat{k} &= -\hat{j}
  \end{align*}
  \]

  These relationships are a result of the “right-hand rule” convention.

  The 3 equations on the left are an example of a “cyclic permutation”
Cross Product – Determinant Form

• Concise way to remember order and sign of terms:
  - Use the **determinant** of a matrix!
  - **Matrices** and **determinants** to be described in detail later
  - For now, just a tool for getting terms and signs right

2 x 2 matrix:
\[
\begin{pmatrix}
a & b \\
c & d
\end{pmatrix}
\]
Determinant:
\[
\begin{vmatrix}
a & b \\
c & d
\end{vmatrix} = ad - bc
\]

3 x 3 matrix:
\[
\begin{pmatrix}
a & b & c \\
d & e & f \\
g & h & i
\end{pmatrix}
\]
Determinant:
\[
\begin{vmatrix}
a & b & c \\
d & e & f \\
g & h & i
\end{vmatrix} = a \det \begin{pmatrix} e & f \\ h & i \end{pmatrix} - b \det \begin{pmatrix} d & f \\ g & i \end{pmatrix} + c \det \begin{pmatrix} d & e \\ g & h \end{pmatrix}
\]

Determinant form of cross product: \[ \vec{A} \times \vec{B} \]
\[
\begin{vmatrix}
\hat{i} & \hat{j} & \hat{k} \\
A_x & A_y & A_z \\
B_x & B_y & B_z
\end{vmatrix}
\]
\[
(A_y B_z - A_z B_y) \hat{i} - (A_x B_z - A_z B_x) \hat{j} + (A_x B_y - A_y B_x) \hat{k}
\]
Dot/Cross Product Examples

• Which of the following makes sense?
  - And in each case, are parentheses necessary?
    \[ \vec{A} \cdot (\vec{B} \times \vec{C}) \quad \vec{A} \times (\vec{B} \cdot \vec{C}) \quad \vec{A} \cdot \vec{B} \cdot \vec{C} \quad \vec{A} \times \vec{B} \times \vec{C} \]

• Imagine a set of N unit vectors such that:
  - 1) the sum of all N vectors is zero
  - 2) the angle between any 2 unit vectors is constant
  - Draw some examples for different N in 2-D and 3-D space
  - Calculate the angle between two vectors in each case