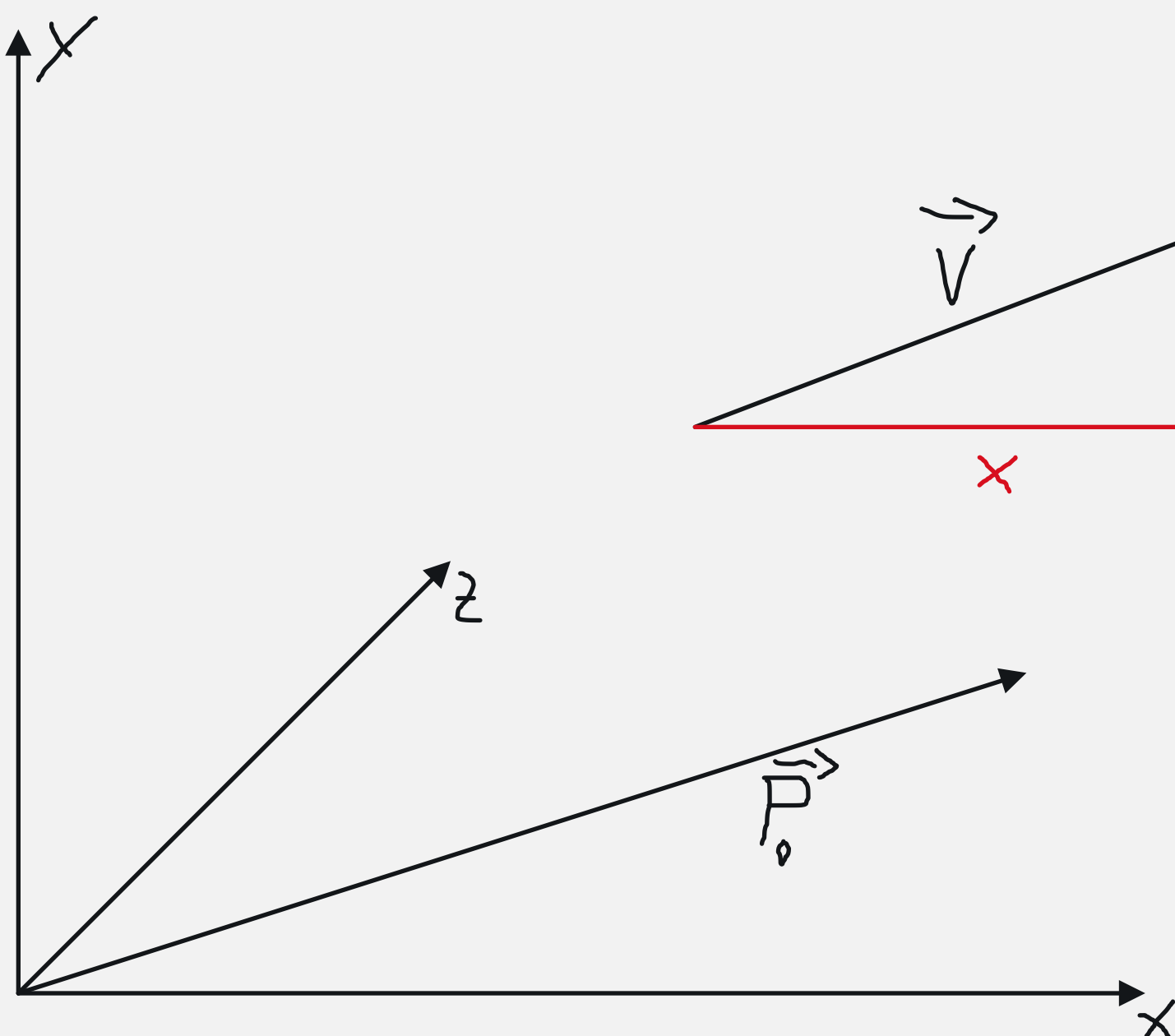


Vectors

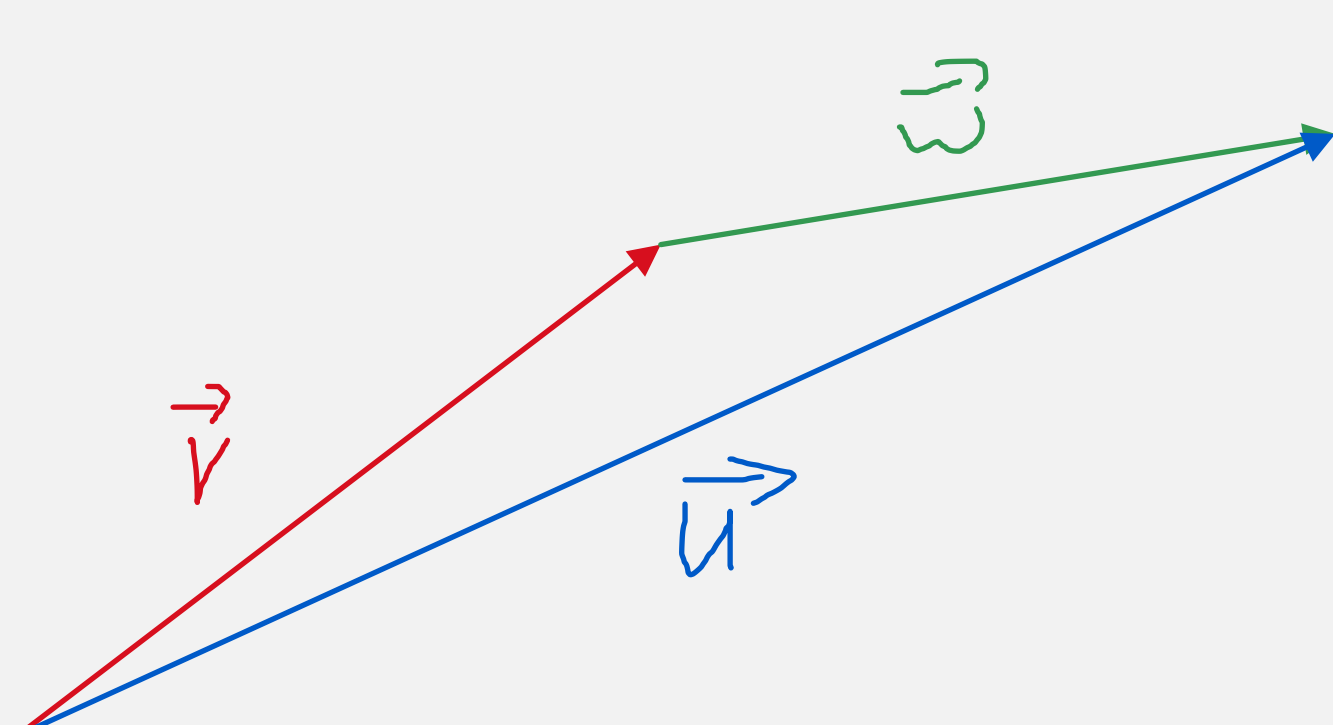


vector - parallel movement  
position vector - cartesian point (vector from the origin)  
direction vector - should always have a length of 1  
unit length vector - vector with the length of 1

$$|\vec{v}| = \sqrt{x^2 + y^2}$$
$$||\vec{v}|| = \frac{\vec{v}}{|\vec{v}|}$$

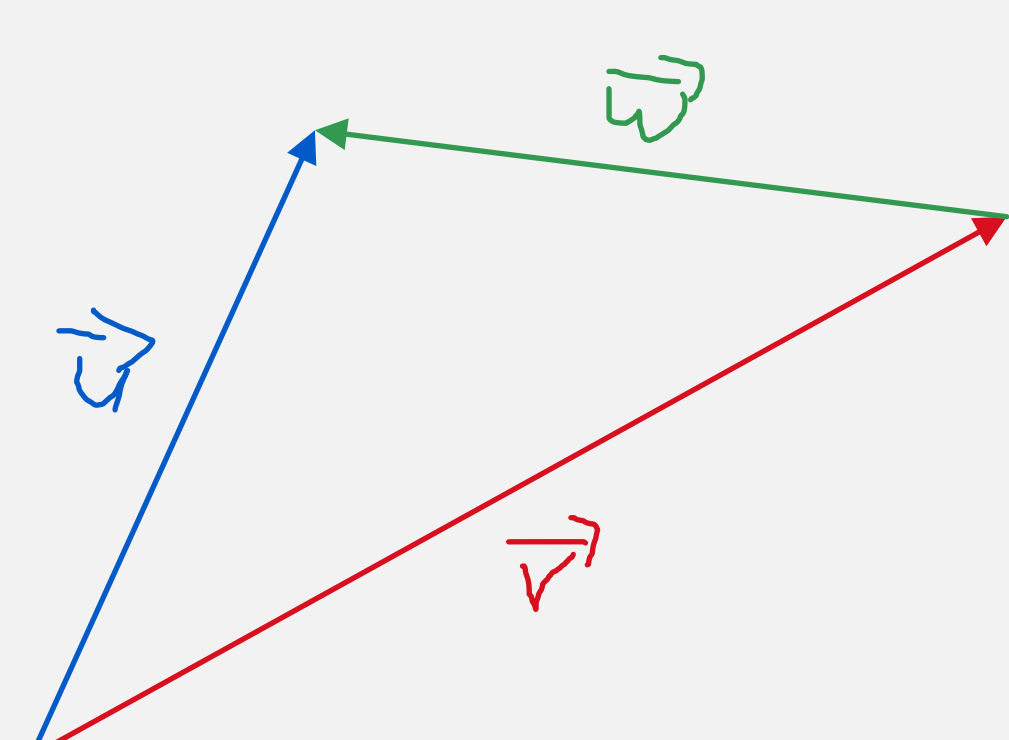
normalized vector

Addition

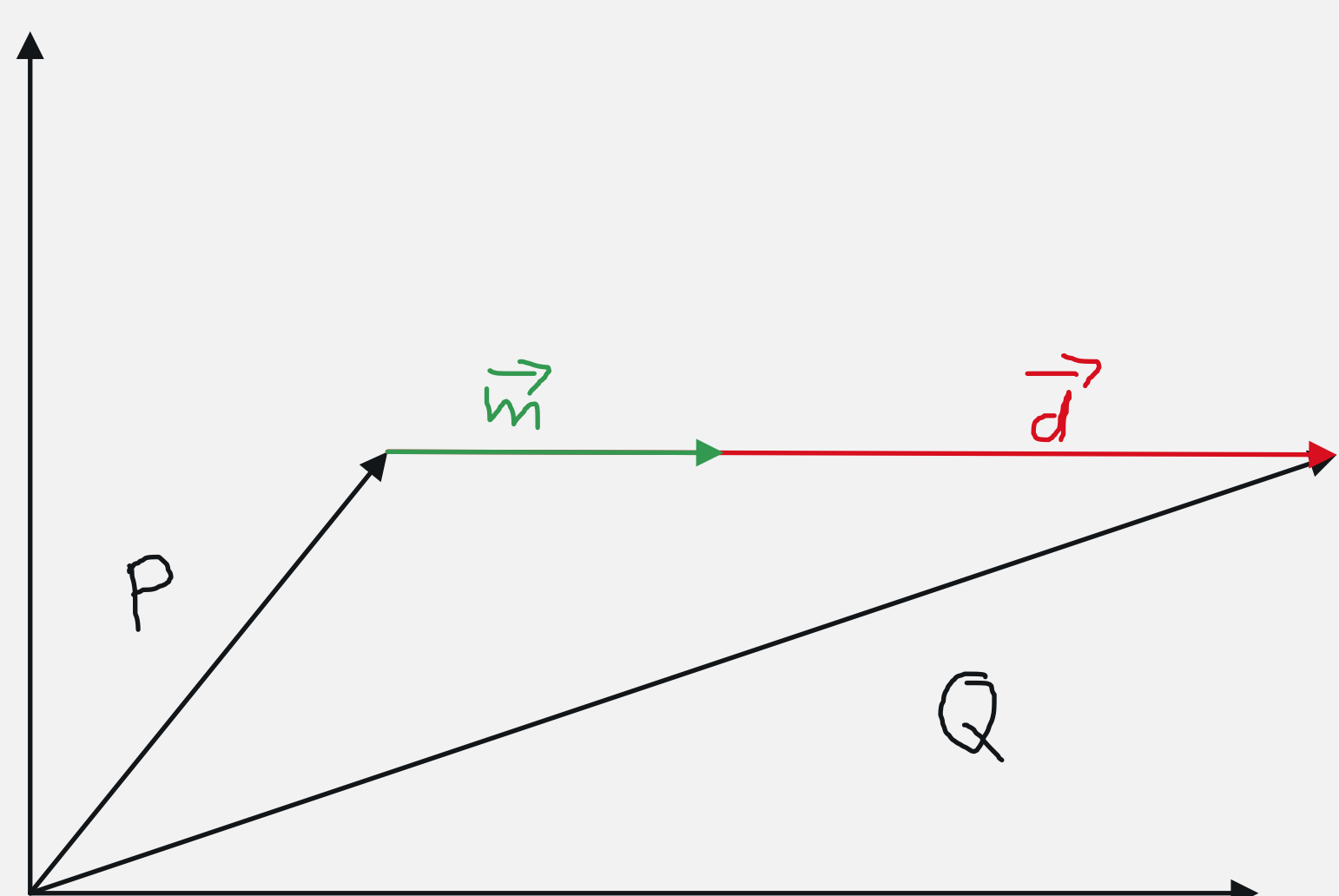


$$\vec{u} = \vec{v} + \vec{w} = \begin{pmatrix} x_v + x_w \\ y_v + y_w \\ z_v + z_w \end{pmatrix}$$

Subtraction



$$\vec{u} = \vec{v} - \vec{w} = \vec{v} + (-\vec{w})$$



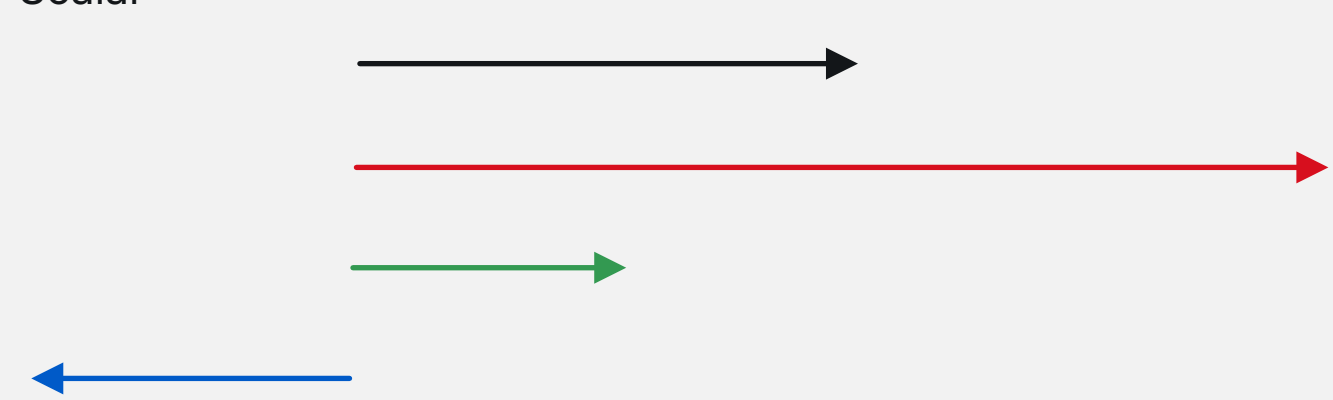
Target minus Start

$$\vec{d} = \vec{Q} - \vec{P}$$

$$\vec{m} = ||\vec{d}|| \cdot (s \cdot dt)$$

movement vector  
s - speed  
dt - delta time

Multiplication with a Scalar

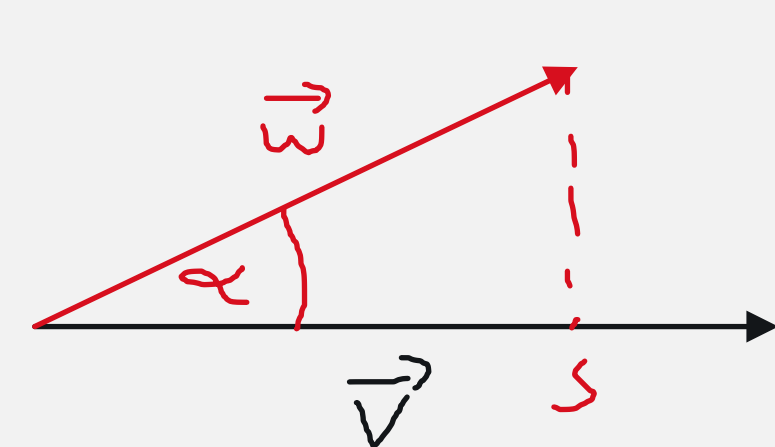


$$\vec{v} \cdot a = a \cdot \vec{v} = \begin{pmatrix} x \cdot a \\ y \cdot a \\ z \cdot a \end{pmatrix}$$

$$\vec{v} \cdot (s \cdot dt)$$

better performance

Dot Product - Scalar Product



$$\vec{v} \cdot \vec{w} = x_v \cdot x_w + y_v \cdot y_w + z_v \cdot z_w$$

$$\vec{v} \cdot \vec{w} = |\vec{v}| \cdot |\vec{w}| \cdot \cos \alpha \Leftrightarrow \cos \alpha = \frac{\vec{v} \cdot \vec{w}}{|\vec{v}| \cdot |\vec{w}|}$$

$$\alpha < \beta \rightarrow \cos \alpha > \cos \beta$$

$$\frac{\alpha}{180^\circ} = \frac{x}{\pi}$$

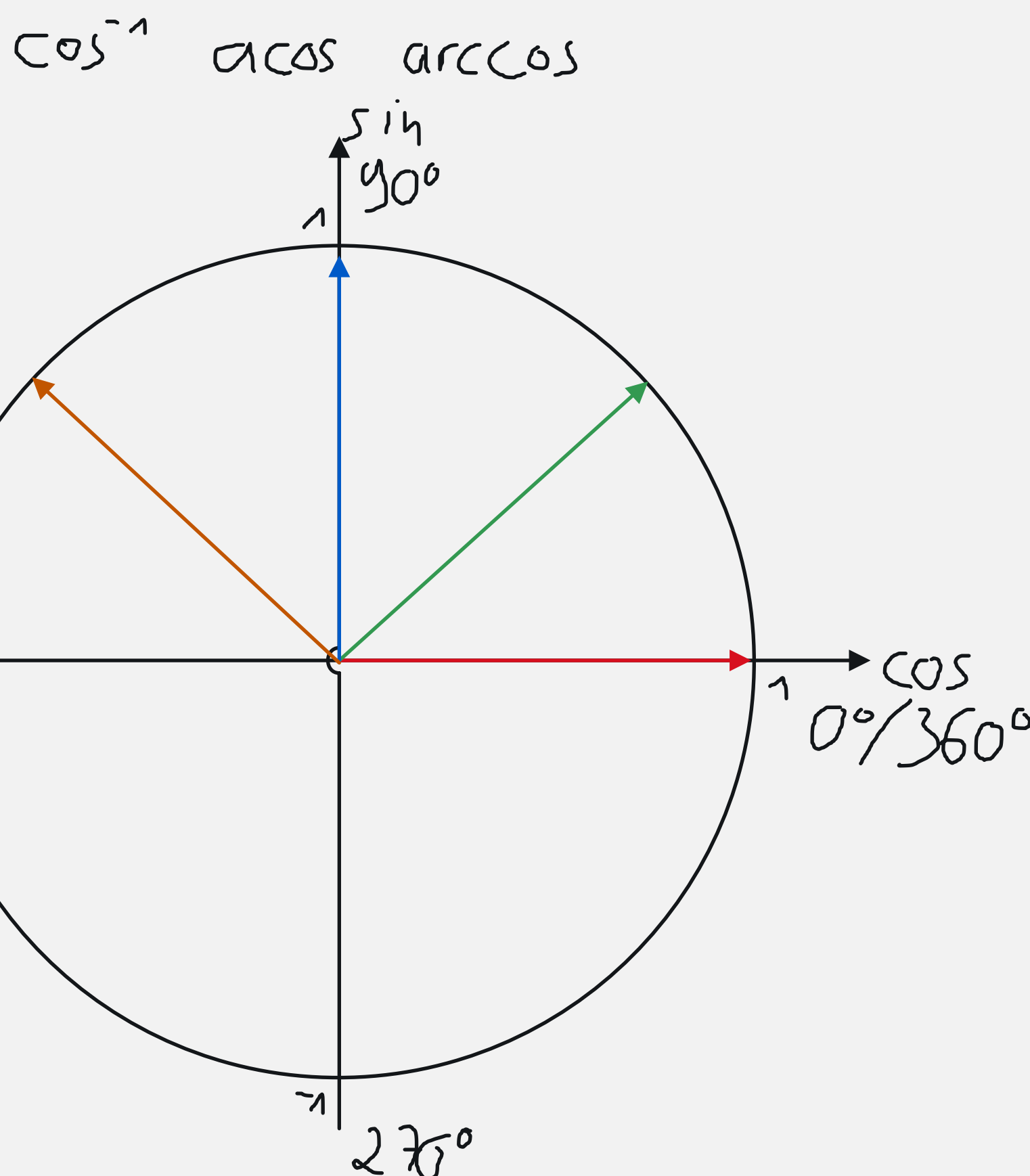
alpha - angle between two vectors  
beta - view angle threshold

$$\alpha = x \cdot \left( \frac{180^\circ}{\pi} \right)$$

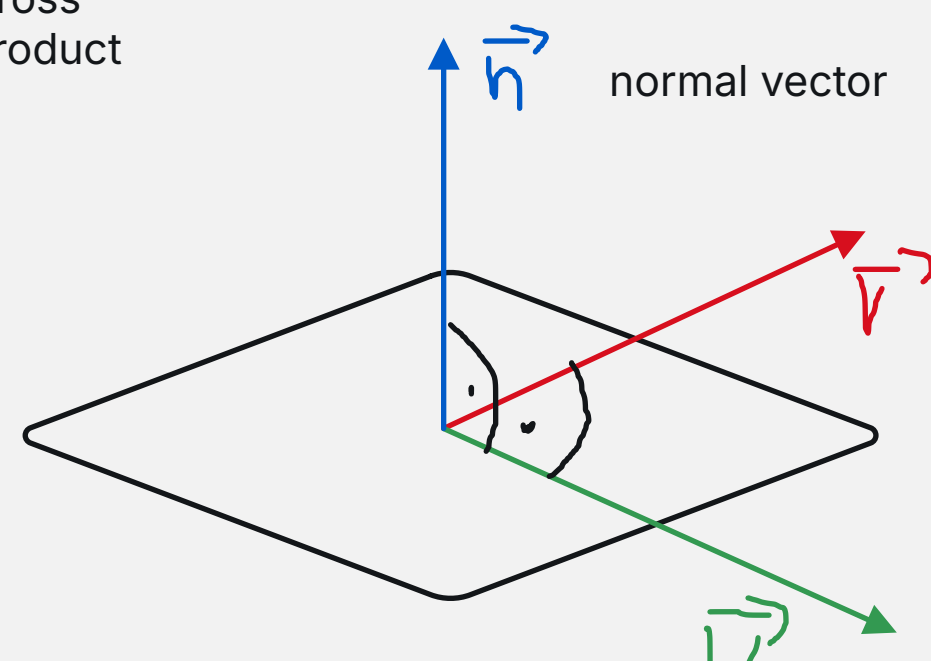
Rad2Deg

$$x = \alpha \cdot \left( \frac{\pi}{180^\circ} \right)$$

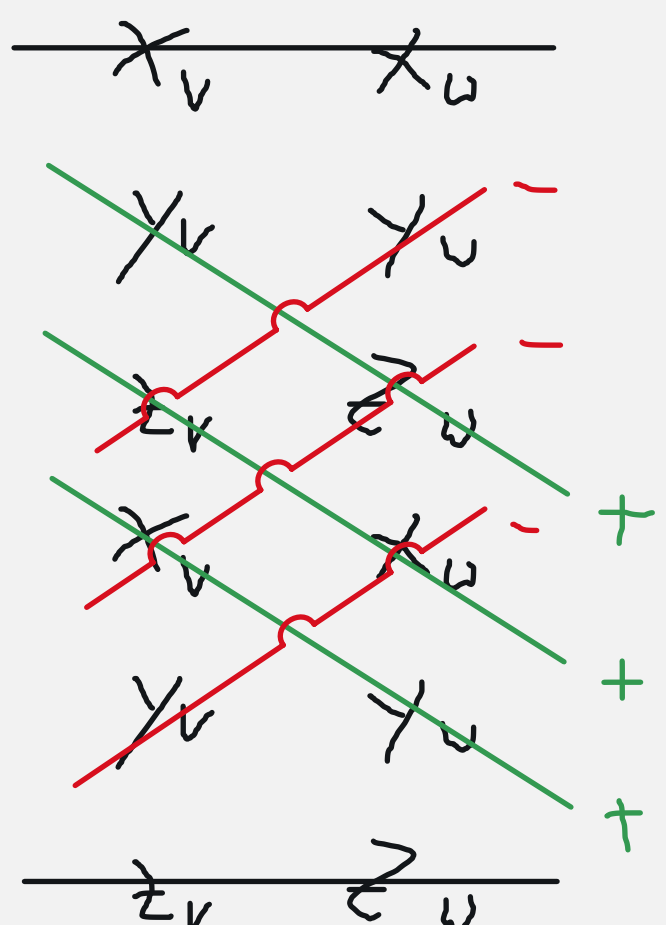
Deg2Rad



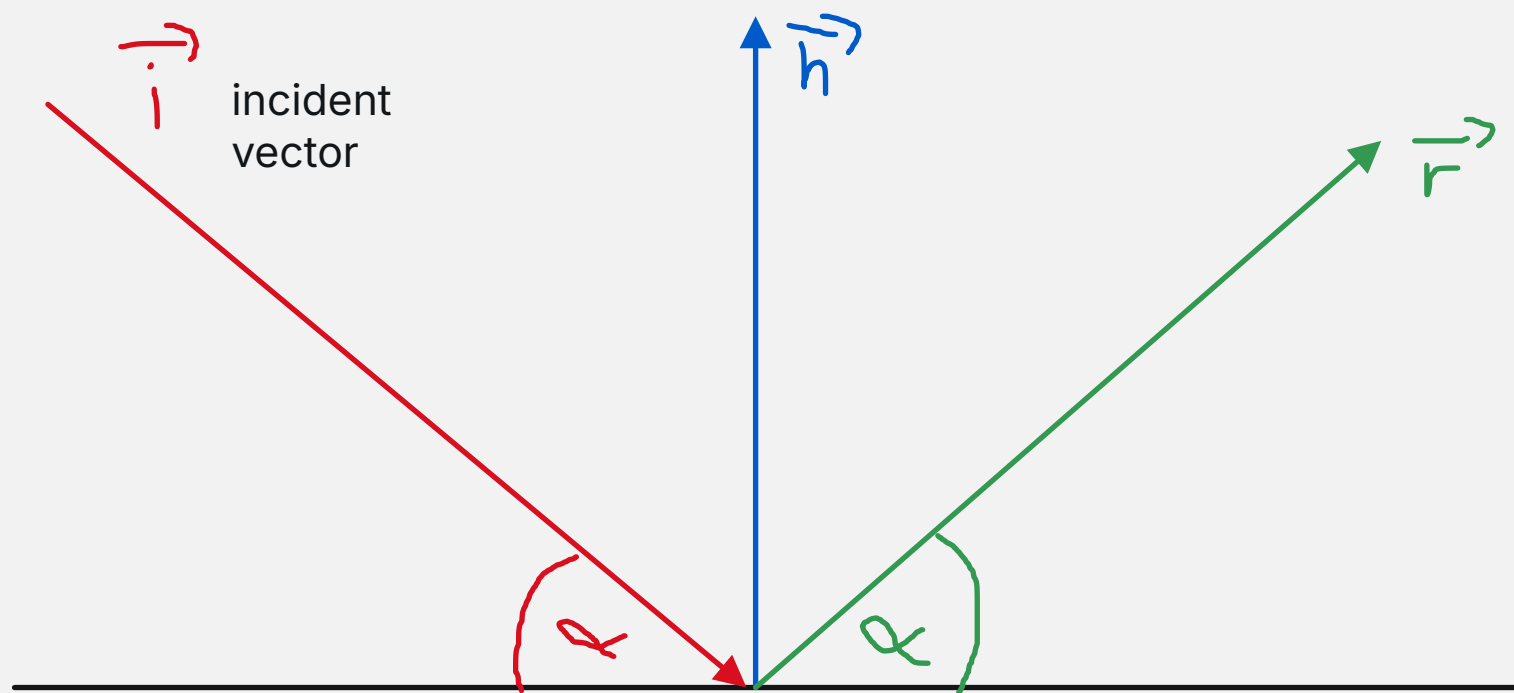
Cross Product



$$\vec{v} \times \vec{w} \neq \vec{w} \times \vec{v} \quad \vec{n} = \vec{v} \times \vec{w} \quad -\vec{n} = \vec{w} \times \vec{v}$$



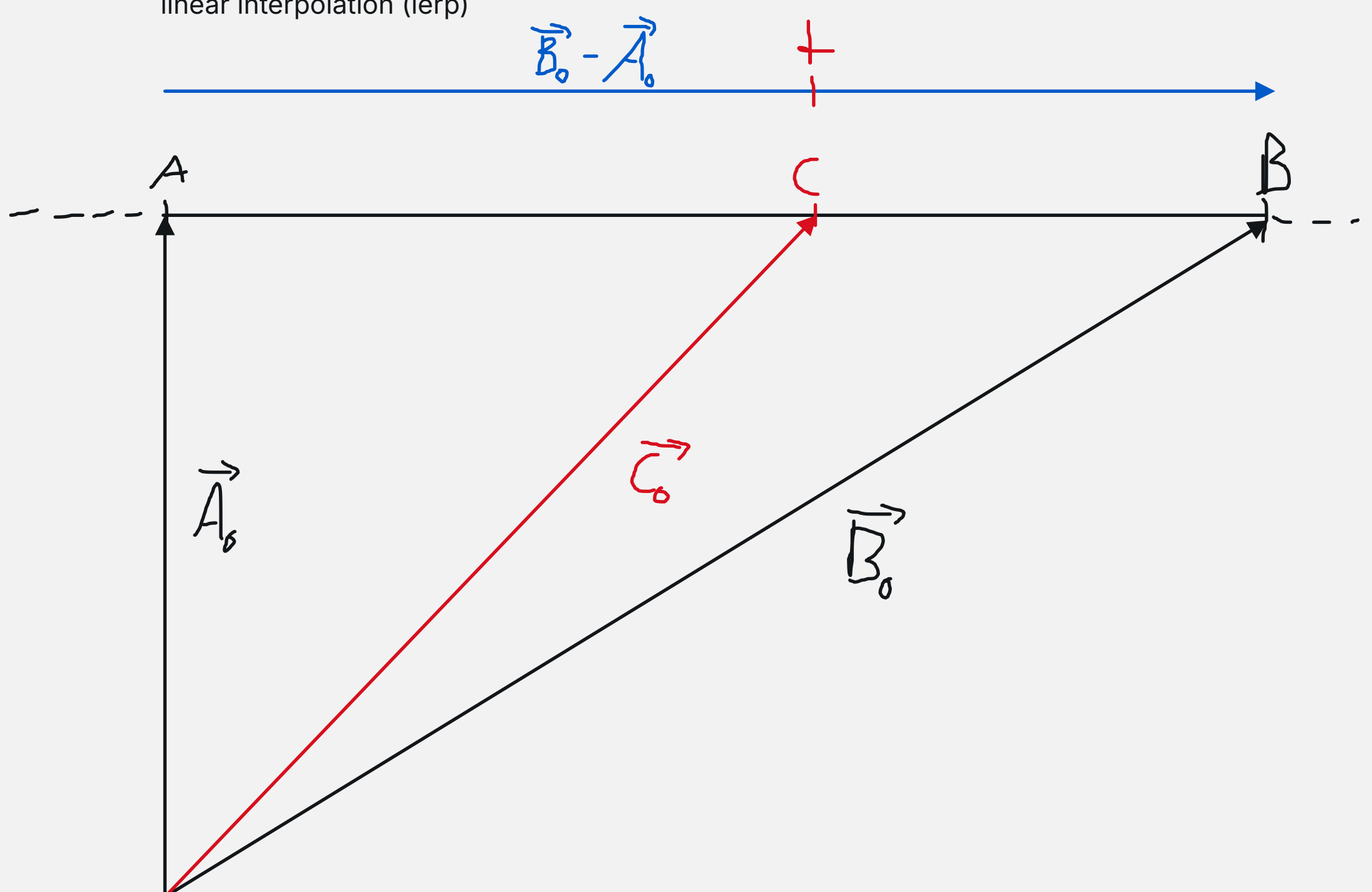
$$\vec{n} = \begin{pmatrix} y_v \cdot z_w - z_v \cdot y_w \\ z_v \cdot x_w - x_v \cdot z_w \\ x_v \cdot y_w - y_v \cdot x_w \end{pmatrix}$$



$$\vec{r} = \vec{i} + 2 \cdot (\vec{n} \cdot \vec{i}) \cdot \vec{n}$$

Interpolations

linear interpolation (lerp)



$$0 \leq t \leq 1$$

$$\vec{C}_t = \vec{A}_0 + t(\vec{B}_0 - \vec{A}_0)$$

t - percentage on the way between A & B

linear extrapolation (lerpUnclamped)  
PingPong

Bezier Curves

first degree - linear



$$B_L(t) = P_0 + t(P_1 - P_0)$$
$$= P_0 + tP_1 - tP_0$$
$$= (1-t)P_0 + tP_1$$

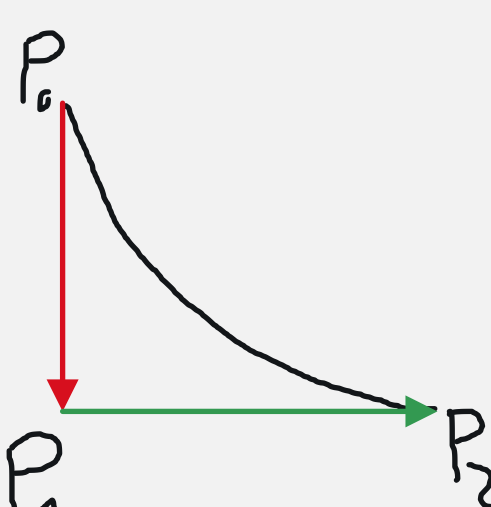
second degree - quadratic

$$B_Q(t) = (1-t)(B_{LP_0P_1}(t)) + t(B_{LP_1P_2}(t))$$

reuse already existing functions

$$= (1-t)^2 P_0 + 2(1-t)t P_1 + t^2 P_2$$

performant version



third degree - cubic

$$B_C(t) = (1-t)(B_{QP_0P_1P_2}(t)) + t(B_{QP_1P_2P_3}(t))$$

reuse already existing functions

$$= (1-t)^3 P_0 + 3(1-t)^2 t P_1 + 3(1-t)t^2 P_2 + t^3 P_3$$

performant version

