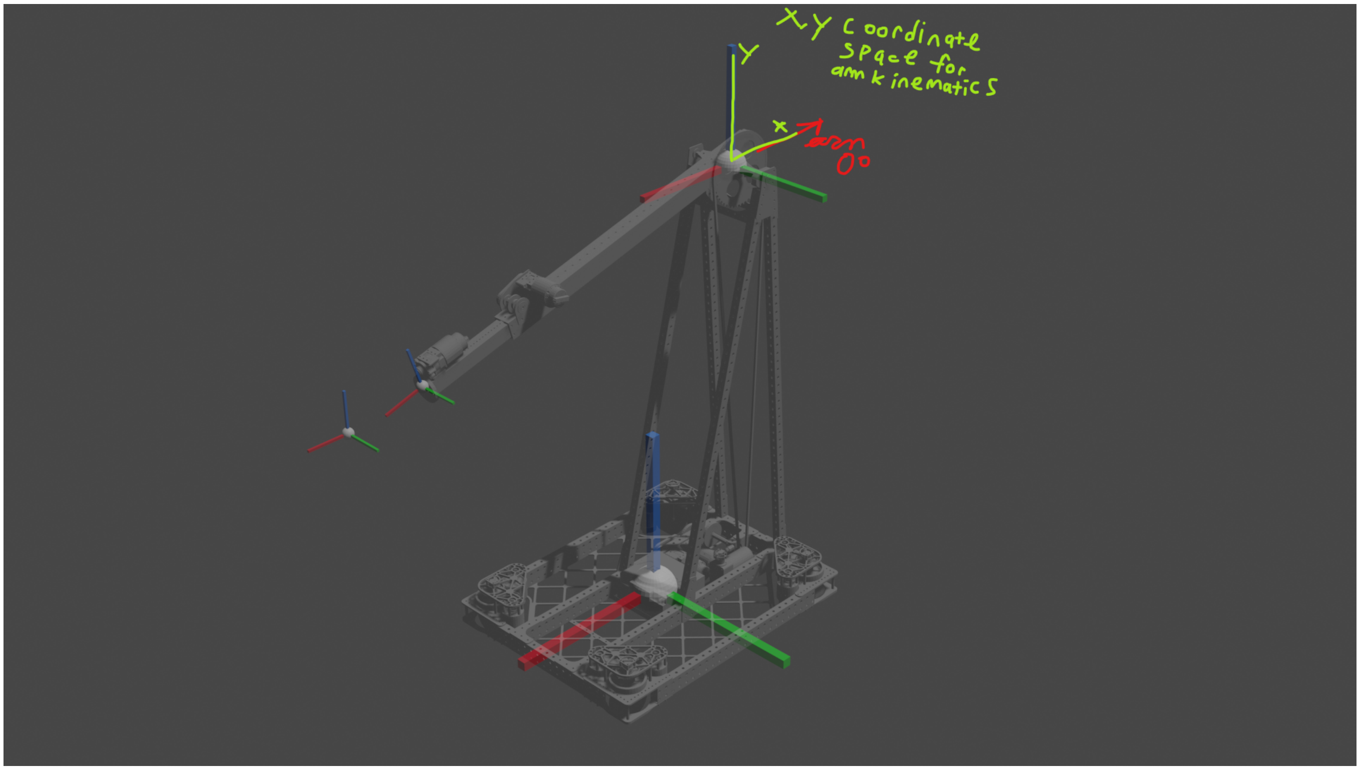
Forward Kinematics

The following is an analytical approach to solving forward kinematics with the robot's arm

The following is an image describing some assumptions about robot coordinate spaces:



> ✎ **Note**
>
> These vectors are all 2d, which makes the math a little simpler to compute. The 2d pose vector can be trasformed into a 3d pose at the end, simply by emplacing values in a 3d vector from the 2d vector. Computation is able to be done in 2d, as the arm system only has 2 DOF

Let L be the current length of the arm

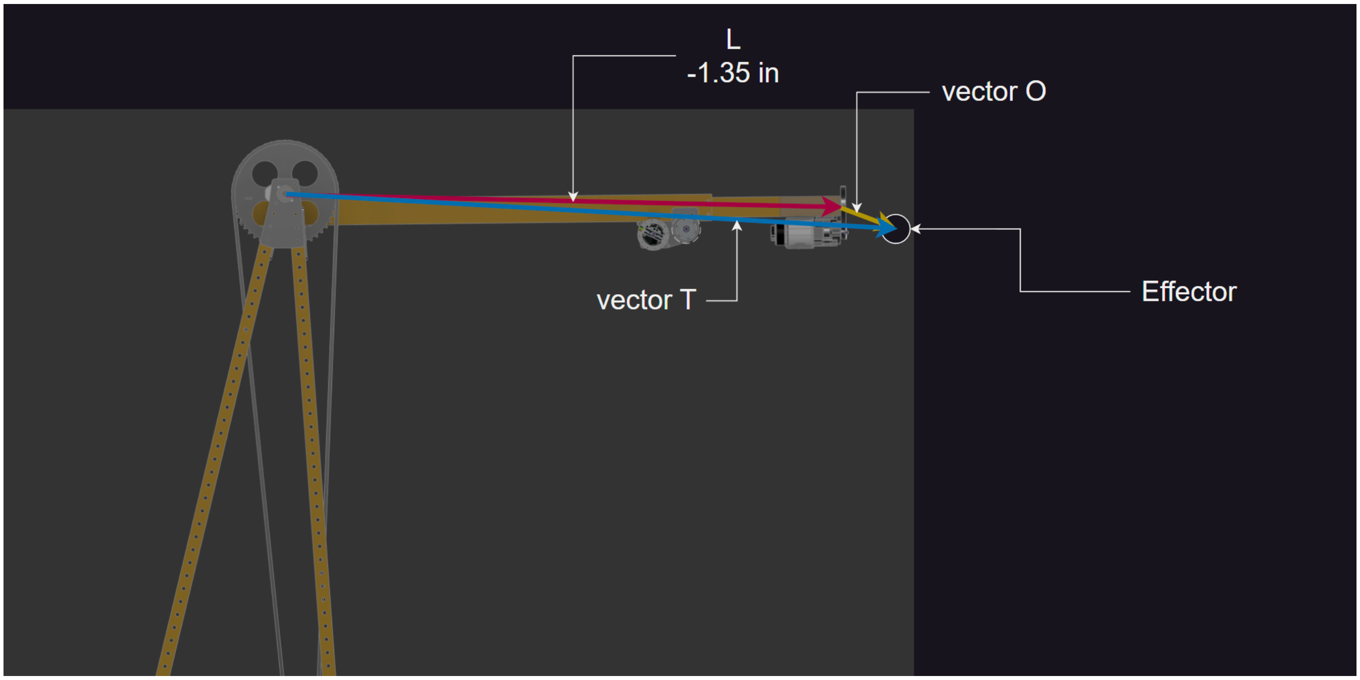
NOTE: $\vec{O}$ assumes the arm is at 0 degrees
Let $\vec{O}$ be the vector representing the transformation for the center of the end effector from the end of the arm.

NOTE: $\vec{T}$ assumes the arm is at 0 degrees
NOTE: -1.35 is the offset in the y axis, in inches, of the arm from the center of rotation when

the arm is at 0°
Let $\vec{T}$ be the vector of the end effector's offset from (0,0) (the center of rotation):

$$\vec{T} = \begin{bmatrix} L \\ -1.35 \end{bmatrix} + \begin{bmatrix} O_1 \\ O_2 \end{bmatrix}$$

The construction of $\vec{T}$ can be visualized as such:



Let θ be the angle of the arm, in radians
Let p be a vector representing the pose of the end effector (IN 2D!), relative to (0,0) which is the rotation point.

$$p = \begin{bmatrix} T_1 \cos\theta - T_2 \sin\theta \\ T_1 \sin\theta + T_2 \cos\theta \end{bmatrix}$$

To transform to arm coordinate system: Let $\vec{p\prime}$ be a 3d vector in the arm coordinate space, it can be constructed as such:

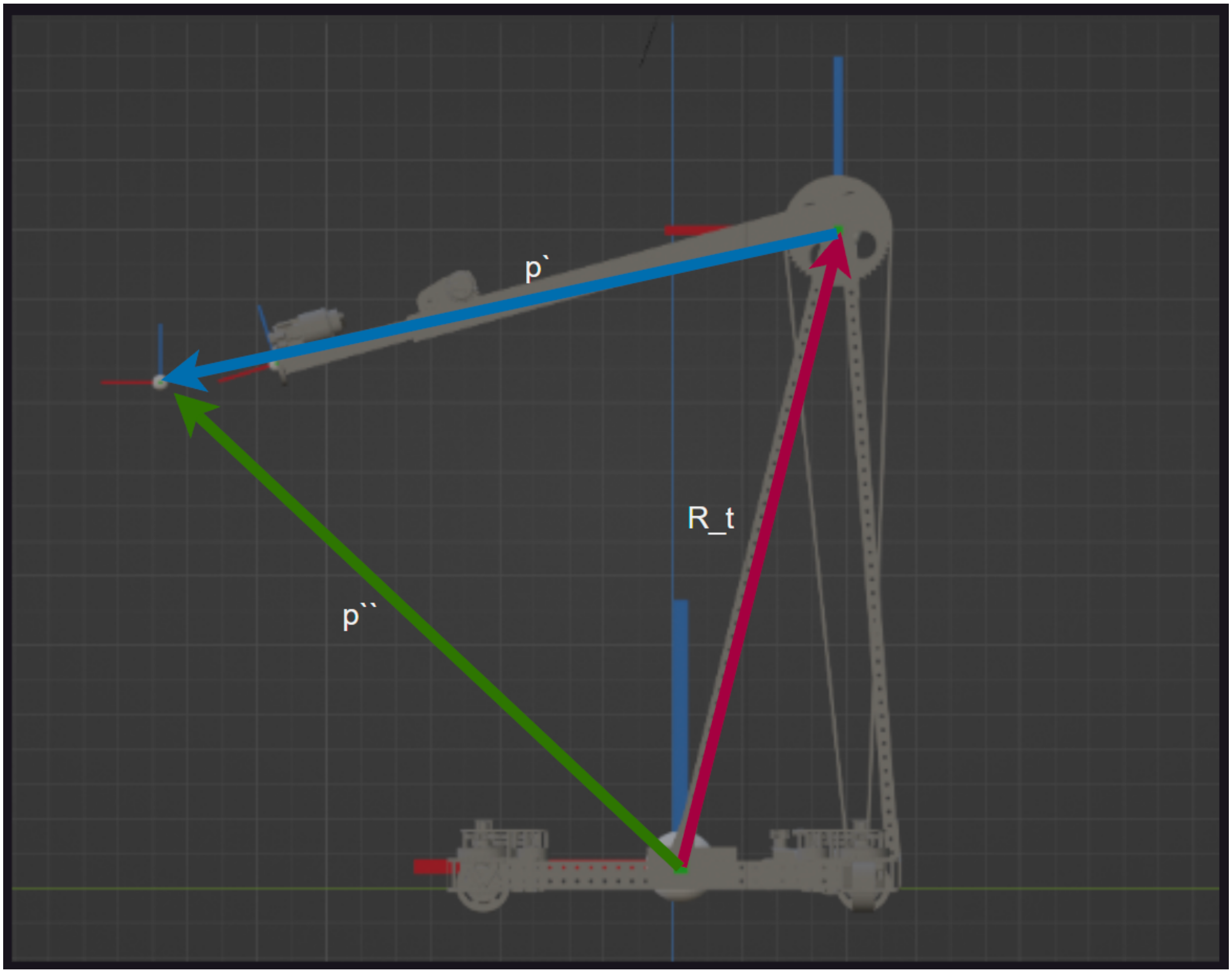
$$p\prime = \begin{bmatrix} -p_1 \\ 0 \\ p_2 \end{bmatrix}$$

Yes, this could be done "*properly*" but this works just fine for our purposes

Finally, to transform into robot space:
Let $\vec{p\prime\prime}$ be the vector representing the end effectors pose in robot coordinate space
Let $\vec{R_t}$ be the vector representing the transformation from robot origin the origin of arm frame

$$p\prime\prime = \vec{R_t} + \vec{p\prime}$$

This is visualized below:



Inverse Kinematics

As per forward kinematics, $\vec{T}$ is still defined as the vector which represents the transformation from the arm origin to the wrist, then to the end effector.

$\vec{O}$ describes the translation from the wrist to the middle of the effector, just like forward kinematics

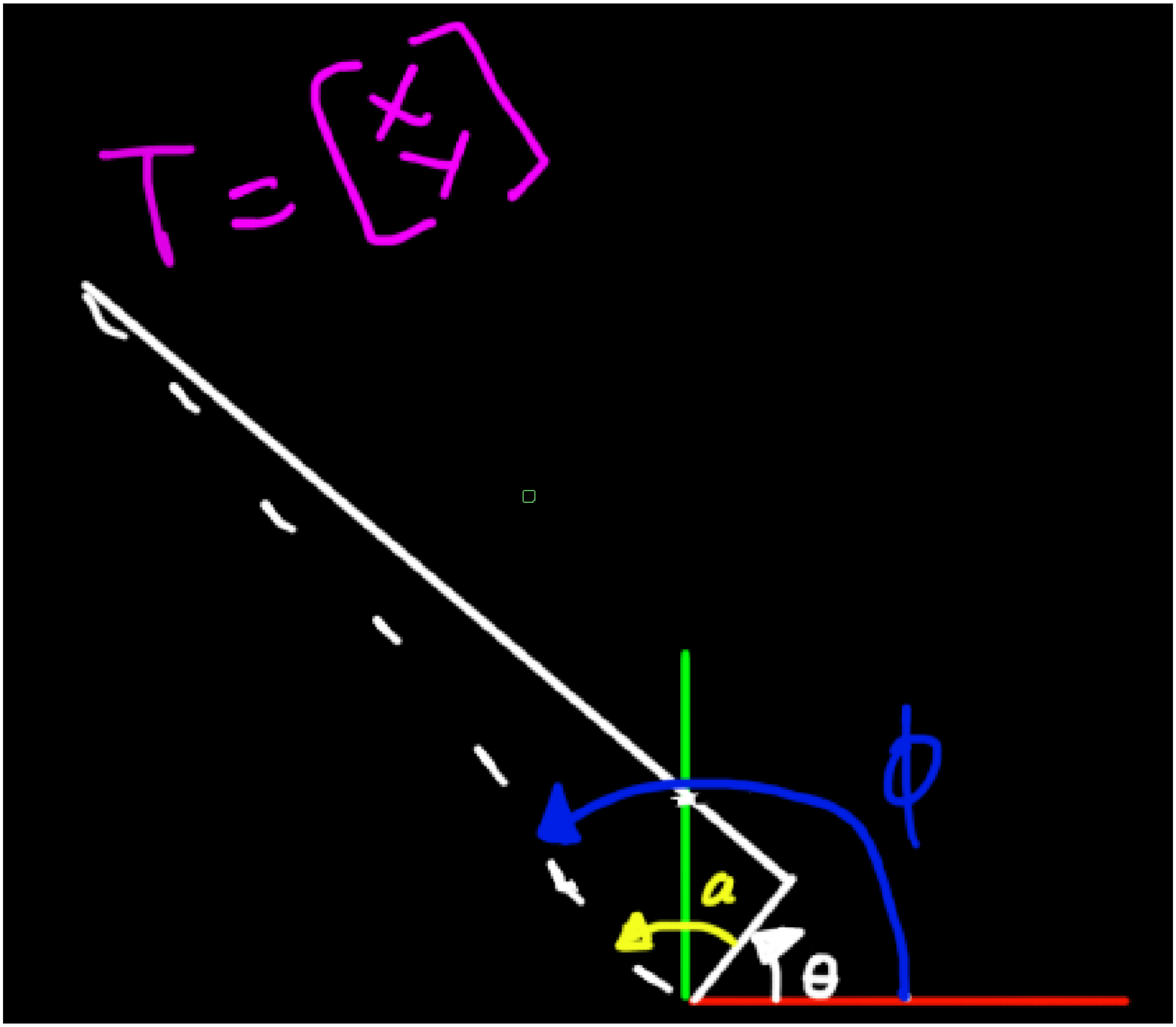
$\vec{T}$ uses two other vectors because the robot may change, so the effector would need to change.

$$\vec{T} = \begin{bmatrix} L \\ -1.35 \end{bmatrix} + \begin{bmatrix} O_x \\ O_y \end{bmatrix}$$

Note

Yes, T_x is variable, but luckly it is unneeded for the IK model, as T_y is the only part of $\vec{T}$ used in the IK model

Define ϕ, α, θ by the following image:



$$\phi = atan2(y, x)$$

$$a = abs(T_y)$$

$$b = \sqrt{||P||^2 - T_y^2}$$

$$\alpha = atan2(b, a) \quad \text{(Assuming atan2 takes y,x)}$$

$$\theta = \phi - \alpha$$

Therefore, the arm angle is equal to:

$$\theta + \frac{\pi}{2}$$

An array containing this data (j) can be represented as:

$$j = \begin{bmatrix} \theta + \frac{\pi}{2} \\ b \end{bmatrix}$$

Note

To get an inverse kinematic model for the arm (no effector), just set the $\vec{O}$ vector to zero