**ES 656:** Human-Robot Interaction Spring 2025

**Activity 4:** Mathematical Modelling of limb to analyze its motion

Shashank Ghosh, Department of Mechanical Engineering
Vishal Ubnare, Department of Cognitive Science

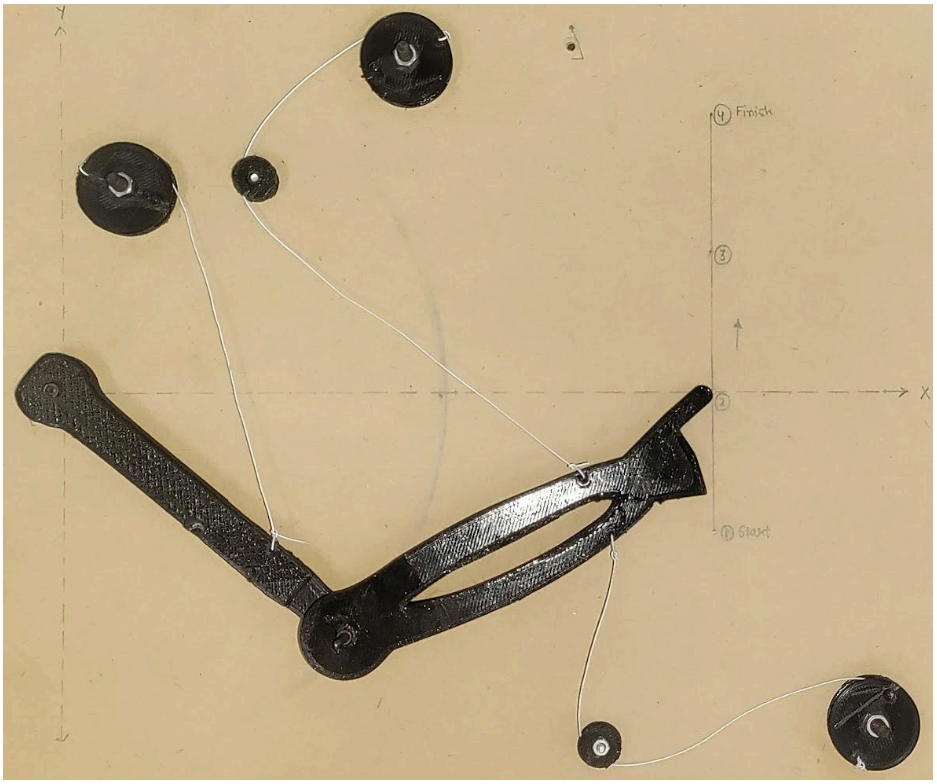



**Fig 1.** Limb model

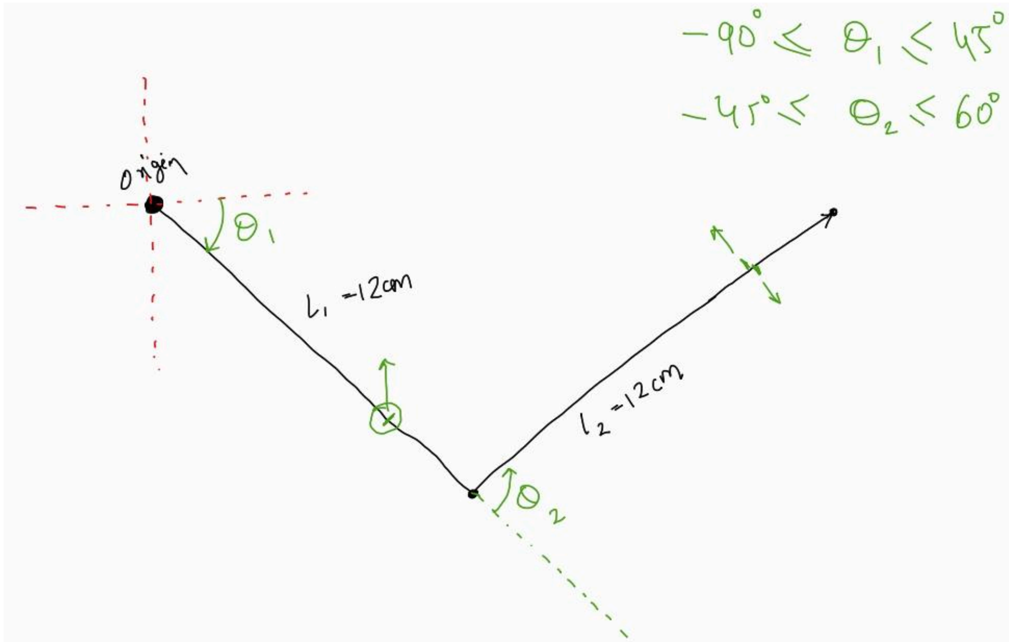



**Fig 2.** Workspace analysis of the Model

For our Model we have a 2R limb with Link lengths as:-

- $L_1$ = 12 cm
- $L_2$ = 12 cm

and due to the physical structure of the model, we have the following constraints on $\theta$ (Joint angles):-

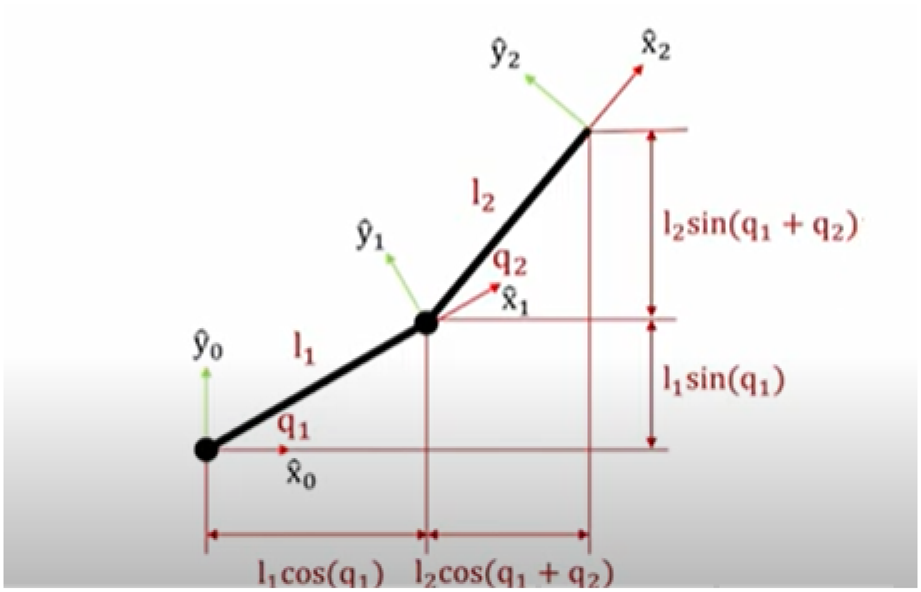
- $-90° \leq \theta_1 \leq 45°$
- $-45° \leq \theta_2 \leq 60°$

**One Redundancy ( m = n+1) , (m=3, n=2)**

One redundancy means it has one actuation method more than required. We usually require only 2 actuation methods for the 2R model but for one redundancy case we have 3 actuation methods (here 3 cables). The additional cables provide extra degrees of freedom in the actuation space, allowing multiple possible tension distributions to achieve the same joint torque.

**Forward Kinematics ( Joint Space to Task Space)**

The forward kinematics of a 2R planar robot (two revolute joints) describes the position of its end-effector (x,y) in a Cartesian plane, given the joint angles ($\theta_1$, $\theta_2$) . For our model,




\

The endpoint of the first link is given by ,

$x = l_1 \cos\theta_1$ , $y = l_1 \sin\theta_1$

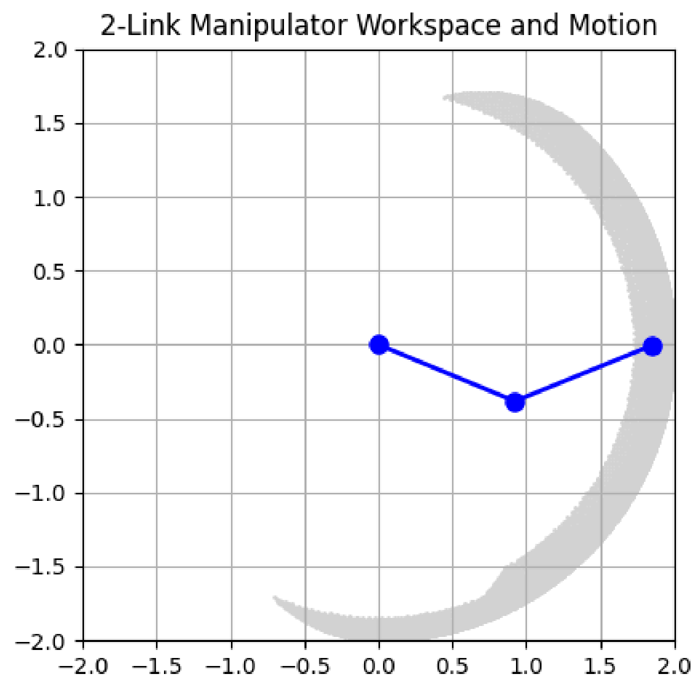
The second link rotates by $\theta_1 + \theta_2$ relative to the X axis, therefore,

$\mathbf{x} = l_2 \cos(\theta_1 + \theta_2)$, $\mathbf{y} = l_2 \sin(\theta_1 + \theta_2)$

Therefore the final end effector position is given by:-

$$\mathbf{X} = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2)$$
$$\mathbf{Y} = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2)$$

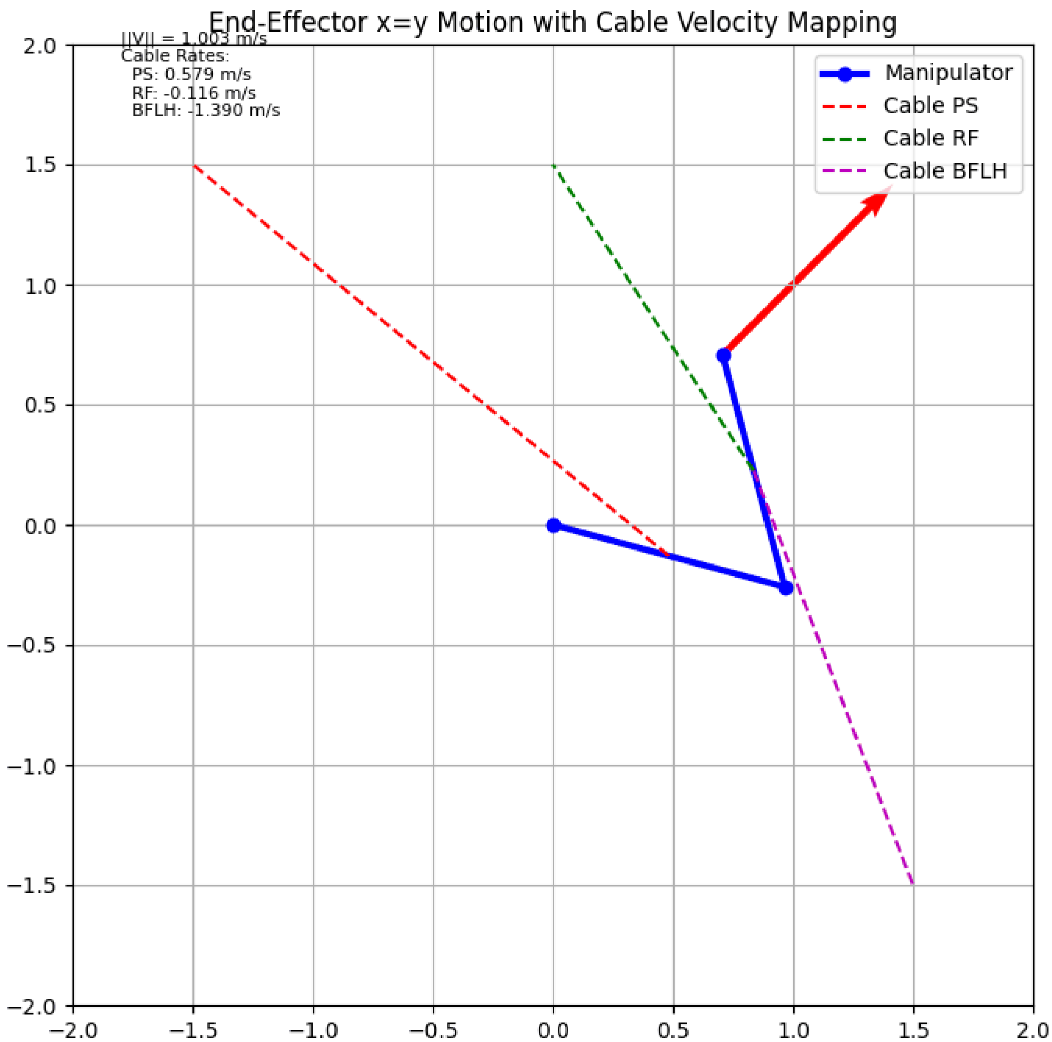
Using this equation we can plot the task space movement.




**Cable space to Joint space to Task space :-**

To perform Cable space to Task space mapping we use the already derived Forward kinematics equation, on top of that we derive cable space to joint space mapping factor.

Below is the mathematical derivation of same, the code to simulate is attached in the repository under the name **1redundancy.py**




Mathematical derivation:-

Each cable length $l$ is a function of joint angles $q = [q_1, q_2]^T$
For small variations, the change in cable lengths is given by :-

$$\dot{l} = J_c \dot{q}$$

Where the joint-to-cable Jacobian $J_c$ is a 3x2 matrix whose i$^{th}$ row is $[\frac{\partial l_i}{\partial q_1}, \frac{\partial l_i}{\partial q_2}]$

The end effector position x is related to joint angles by

$$x = g(q),$$

So that its velocity is :-

$$\boldsymbol{v} = \dot{x} = J_t \dot{q}$$

where $J_t$ is 2x2 task-space Jacobian derived earlier.

Using the above two equations we can relate $\dot{l}$ and $v$ .

Substituting,

$$\dot{l} = J_c J_t^{-1} \boldsymbol{v}$$

Using this we can visualise 1 redundancy case and 2 redundancy cases easily.

For 2 redundancy case the $J_c$ would be slightly different.

# References


1. Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2006). *Robot Modeling and Control*. Wiley.
2. Chen, W. H., & Agrawal, S. K. (2012). Optimal cable routing and tension distribution for a cable-driven robot using particle swarm optimization. *ASME Journal of Mechanical Design*, 134(9), 091005.
3. Zhang, Y., Zhang, D., & Gao, F. (2020). Workspace‑based cable routing optimization of cable‑driven parallel robots. *Mechanism and Machine Theory*, 147, 103767.