Robotics

Trajectory Planning

Moving in a Line

Big Question: How does one command the robot's end-effector to move in a line?

An intuitive approach would involve way points between the line's start and end points. Using inverse kinematics, the manipulator's joint angles for each way point can be calculated. Moving to those way points would trace out a line.

Recall that the parametric equation of a line $\boldsymbol{l}(t)$ has form $\boldsymbol{l}(t) = \boldsymbol{p} + t\boldsymbol{v}$. If say that $\boldsymbol{v} = \boldsymbol{q} - \boldsymbol{p}$ then

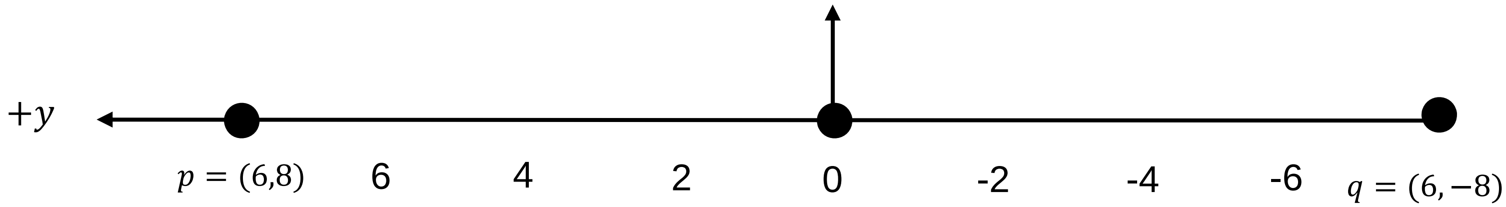
$$\boldsymbol{l}(t) = \boldsymbol{p} + t\boldsymbol{v} = \boldsymbol{p} + t(\boldsymbol{q} - \boldsymbol{p}) = \boldsymbol{p}(1 - t) + t\boldsymbol{q}$$

Thus, to have n equally spaced points on line segment between p and q, then define n values of $t \in [0,1]$ as

$$t = \frac{i}{n} \quad \text{where } \text{i}=0, 1, \ldots n - 1$$

Problem: For our LEGO 2-link manipulator, recall that 1-stud Bricks were placed at start point $p = (x_o, y_o) = (6, 8)$ and end point $q = (6, -8)$. Suppose one wants 1 way point in the middle of the starting and ending points. What is the location of that way point?

Solution: Visually, one sees that the way point is (6, 0)



The parametric equation of a line confirms this. One has 2 equally spaced segments. Hence:

i	*t*	*l*
0	0	8
1	**0.5**	**0**
2	1	-8

$$t = \frac{1}{2} = 0.5$$

and when $i = 1$ then $t = \frac{1}{2} = 0.5$

and $l\left(\frac{1}{2}\right) = 8(1 - 0.5) + 0.5 \times (-8) = 4 - 4 = 0$

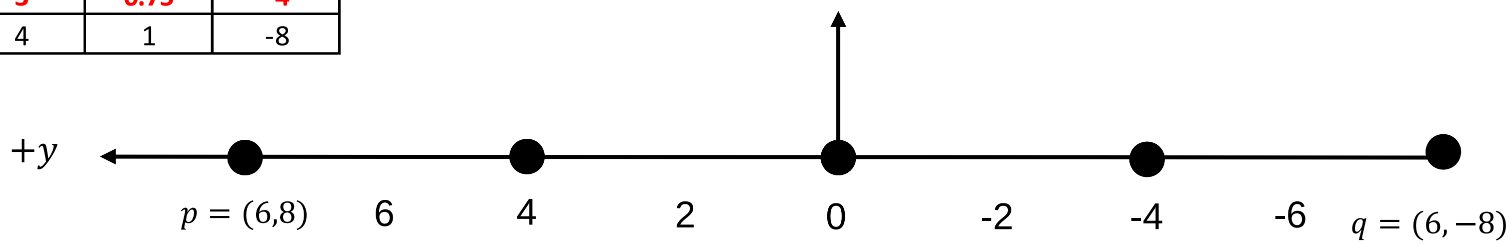
Problem: Suppose want **3** points between start and end points? Hence **4** equally spaced segments. What are the locations of the way points?

Solution: By inspection should have $\mathrm{n} = 4$ hence $t = \frac{i}{4}$ where i=0, 1, ... $n - 1$

i	*t*	*l*
0	0	8
1	**0.25**	**4**
2	**0.5**	**0**
3	**0.75**	**-4**
4	1	-8

If $i = 1$ then $t = \frac{1}{4} = 0.25$ and $l\left(\frac{1}{4}\right) = 8(1 - 0.25) + 0.25 \times (-8) = 6 - 2 = 4$

Visually one has:



Velocity- versus Position-based Control

The way point approach is essentially position-based control; the end-effector (possibly attempts to) visit each point. This leads to a discrete (go-stop-go) motion rather than a continuously smooth one.

Another way to view this is to ask "What is needed if one wants to move the end-effector from points p to q in t seconds?" This is difficult to perform without velocity-based control.

Answering this question is the field of trajectory, motion and path planning. It is further complicated when obstacles are between the start and end points.

Motion Planning is beyond the scope of this course. To pursue this, some key words in the robot course literature include:

- Trapezoidal motion profiles (also called cubic polynomials)
- S-curve motion profiles (also called 5^{th} order or quintic polynomials)
- B-splines

Beyond trajectory generation, a robot's **Jacobian** plays a critical role in motion control and maps joint velocities (and consequently accelerations and forces/torques) to task space velocities.