ROBOTIC FUNDAMENTALS (UFMF4X-15-M)

Trajectories

(Cartesian Space)

UWE Bristol University of the West of England

Previously on ROBOTIC FUNDAMENTALS

Joint interpolated movement is simple to implement but does not provide straight line motion.

Linear or polynomial trajectories can be calculated.

Polynomial trajectories can keep ‘jerk’ low.

Continuous path motion gives a “smoother” motion.

Questions?

Today’s Lecture

Trajectories with Vias

Cartesian Space

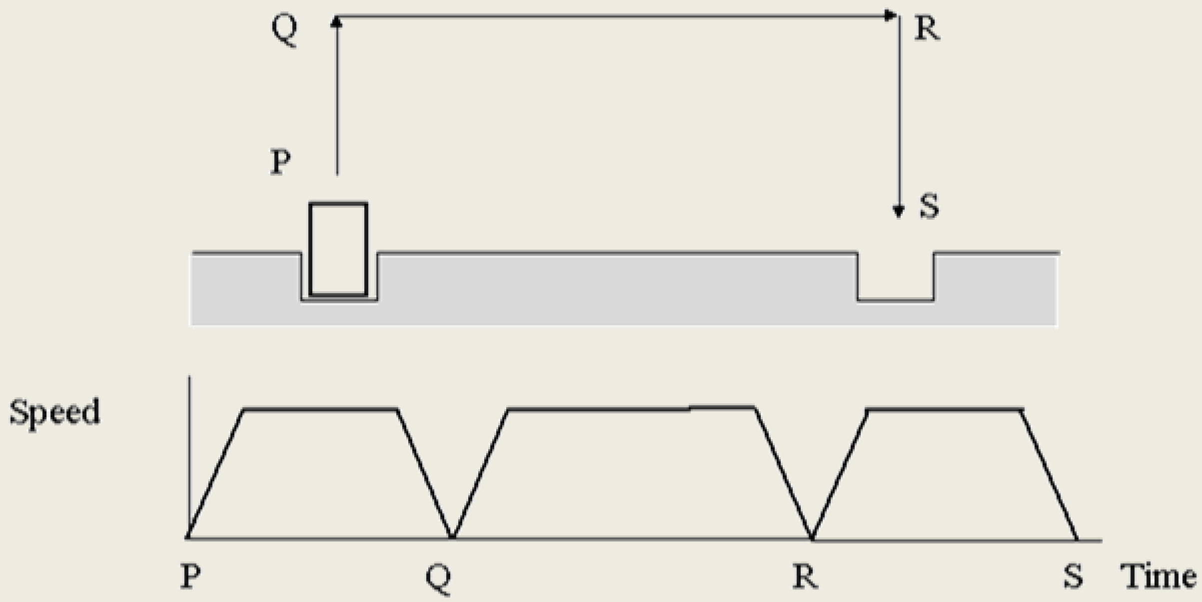
Outline

- Via Points
- Path Generation – Cartesian Space
  - Difficulties / Constrains
  - Linear Motion
- Run Time Path Generation
- Collision-free Path Planning

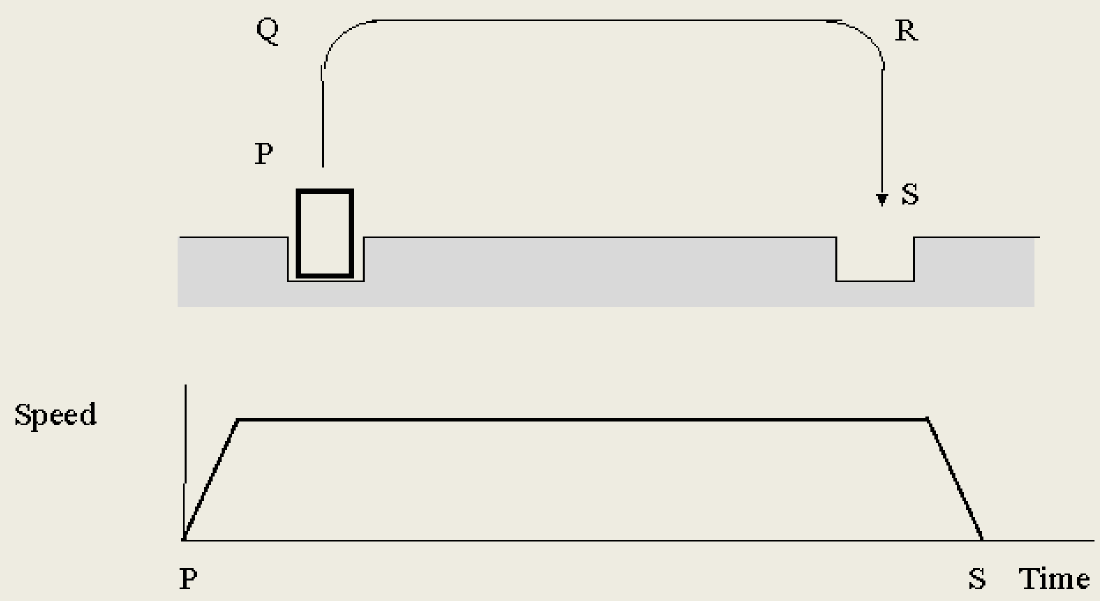
Joint Space Trajectories

VIA POINTS

Non Continuous Path Mode



Continuous Path Mode



Path motion with via points

Why use via points:

- As the order of polynomial increases, its oscillatory behaviour also increases
- Numerical accuracy decreases with the increased order polynomial
- Coefficients have to be recomputed if only one point on the trajectory changes

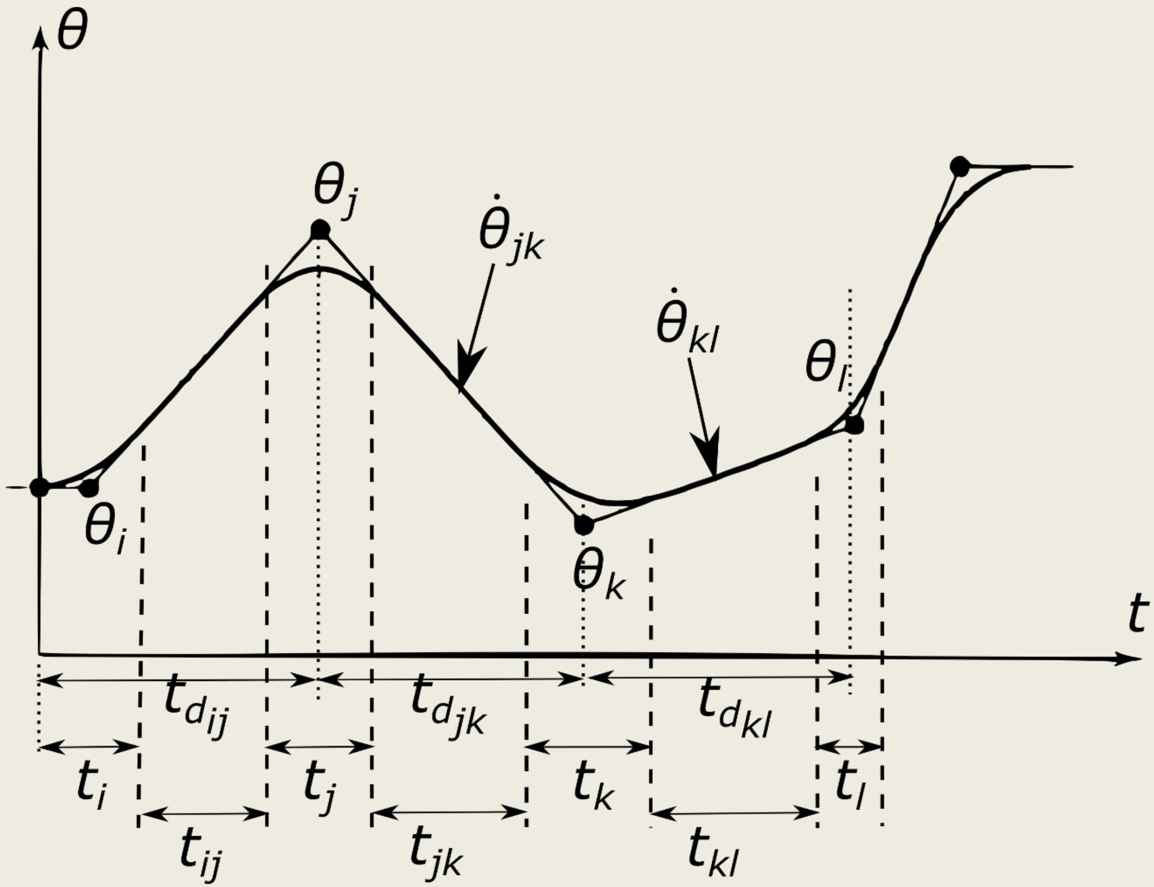
Via points described in terms of a desired position and orientation

Low order polynomials connect the via points

Velocity constraints are not zero in via points

Velocity in via points chosen in a way to maintain constant acceleration

Via points – For a single joint parameter

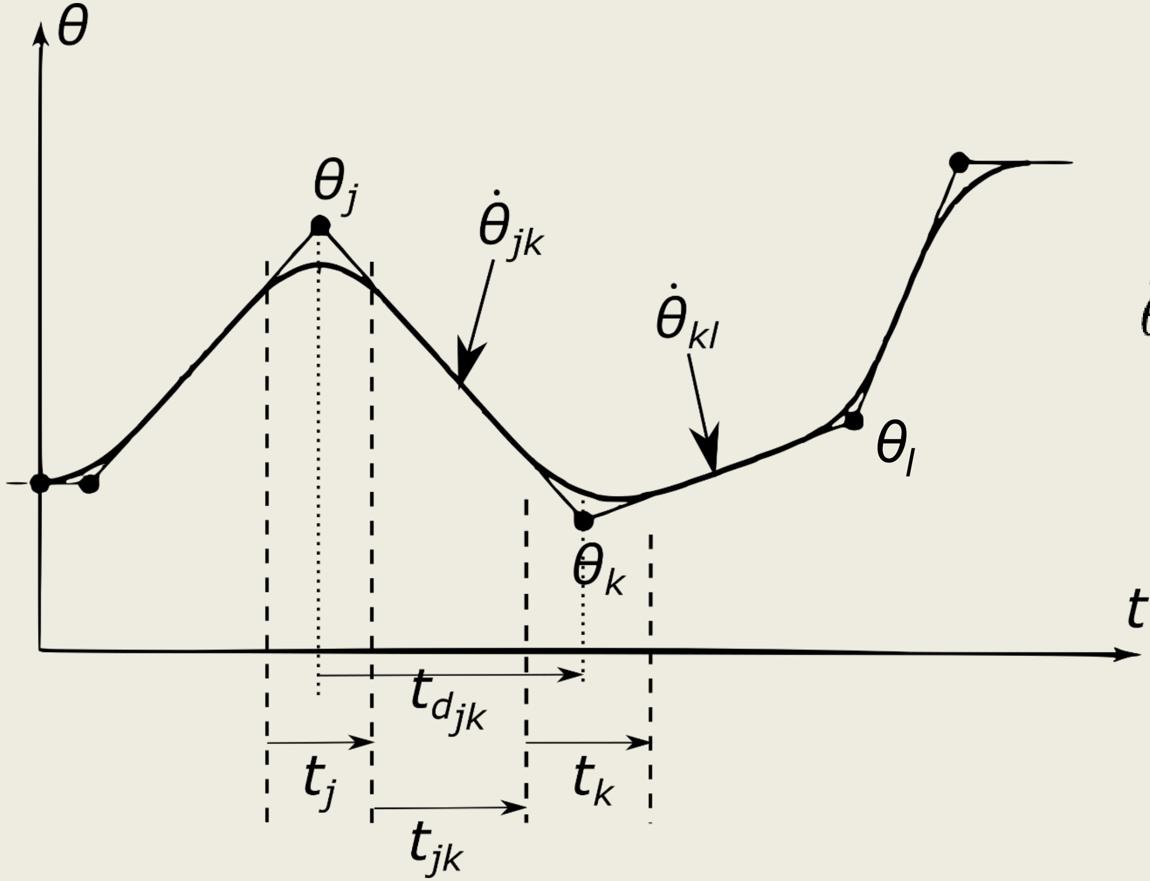


$t_i\ t_j\ t_k\ t_l$: blend times

$t_{ij}\ t_{jk}\ t_{kl}$: linear times

$t_{dij}\ t_{djk}\ t_{dkl}$: durations

Via points – Middle Sections



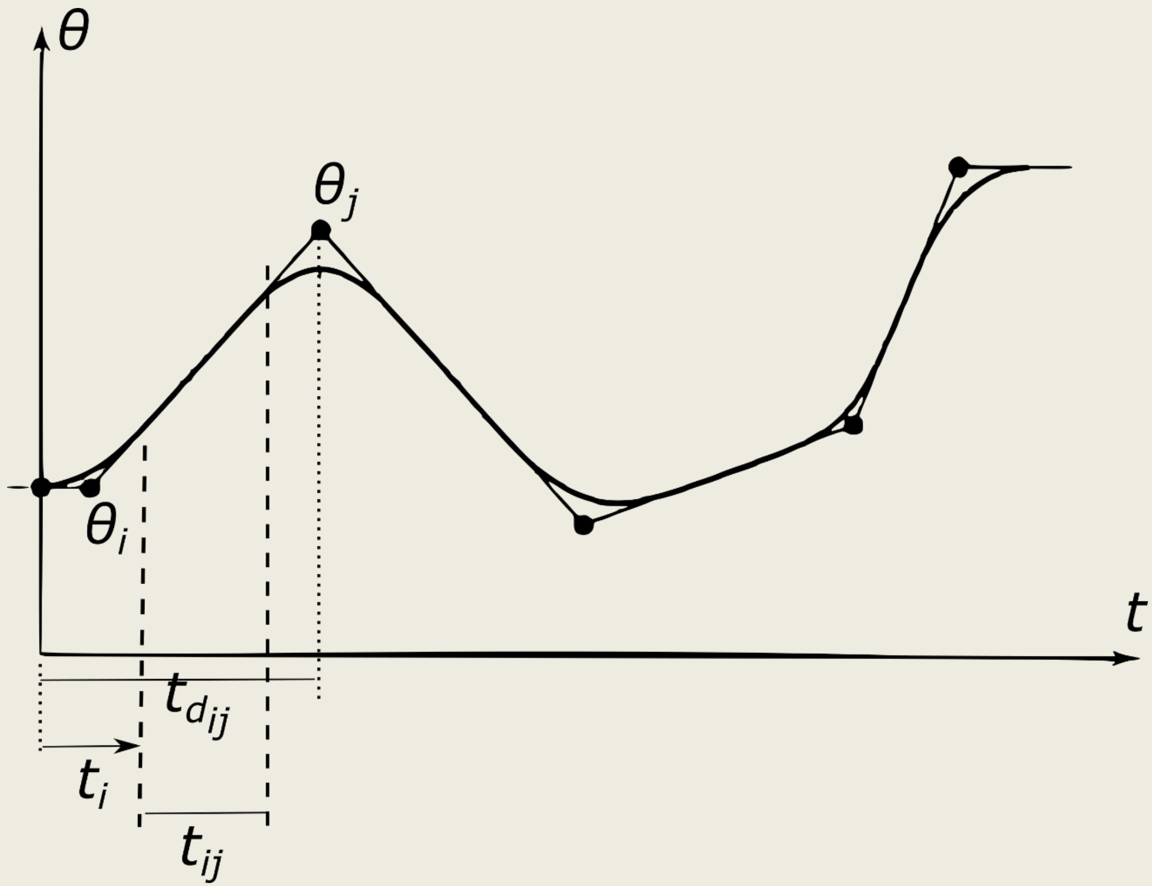
$$\dot{\theta}_{jk} = \frac{\theta_k - \theta_j}{t_{djk}}$$

$$\ddot{\theta}_k = SGN(\dot{\theta}_{kl} - \dot{\theta}_{jk})\left|\ddot{\theta}_k\right|$$

$$t_k = \frac{\dot{\theta}_{kl} - \dot{\theta}_{jk}}{\ddot{\theta}_k}$$

$$t_{jk} = t_{djk} - \frac{1}{2}t_j - \frac{1}{2}t_k$$

Via points – Starting Section



$$\ddot{\theta}_1 t_1 = \frac{\theta_2 - \theta_1}{t_{d_{12}} - \frac{1}{2}t_1}$$

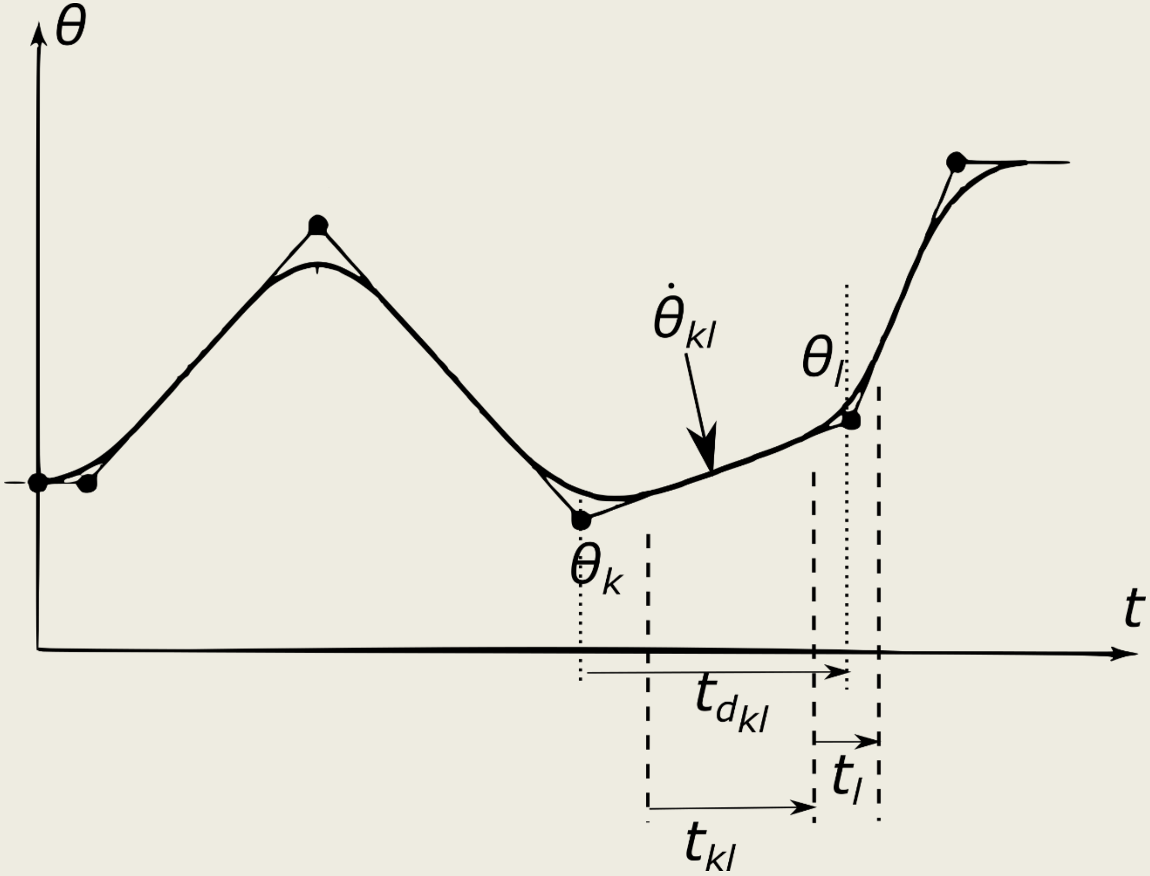
$$\ddot{\theta}_1 = SGN(\theta_2 - \theta_1)\left|\ddot{\theta}_1\right|$$

$$t_1 = t_{d12} - \sqrt{t_{d_{12}}^2 - \frac{2(\theta_2 - \theta_1)}{\ddot{\theta}_1}}$$

$$\dot{\theta}_{12} = \frac{\theta_2 - \theta_1}{t_{d_{12}} - \frac{1}{2}t_1}$$

$$t_{12} = t_{d_{12}} - t_1 - \frac{1}{2}t_2$$

Via points – Finishing Section



$$\ddot{\theta}_l t_l = \frac{\theta_k - \theta_l}{t_{d_{kl}} - \frac{1}{2}t_l}$$

$$\ddot{\theta}_l = SGN(\theta_k - \theta_l)\left|\ddot{\theta}_l\right|$$

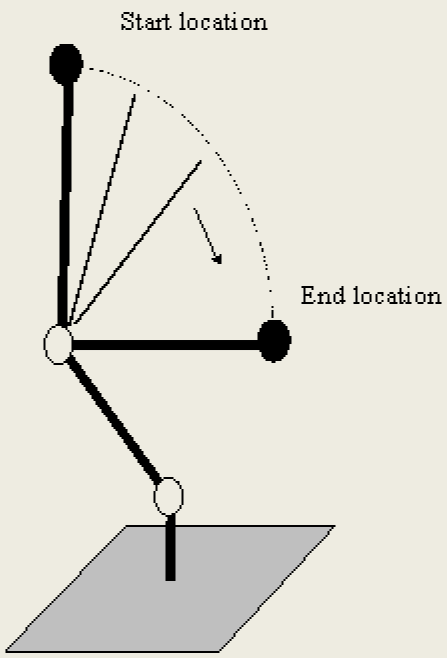
$$t_l = t_{dkl} - \sqrt{t_{d_{kl}}^2 - \frac{2(\theta_l - \theta_k)}{\ddot{\theta}_l}}$$

$$\dot{\theta}_{kl} = \frac{\theta_l - \theta_i}{t_{d_{kl}} - \frac{1}{2}t_l}$$

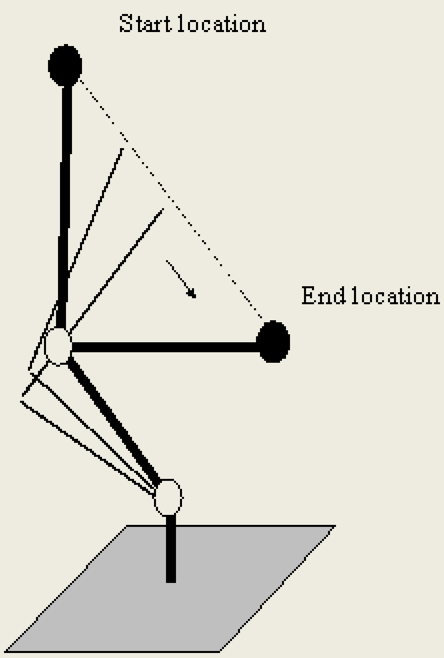
$$t_{kl} = t_{d_{kl}} - t_l - \frac{1}{2}t_k$$

CARTESIAN SPACE TRAJECTORIES

Cartesian Space Trajectories



Joint-interpolated movement



Straight line movement

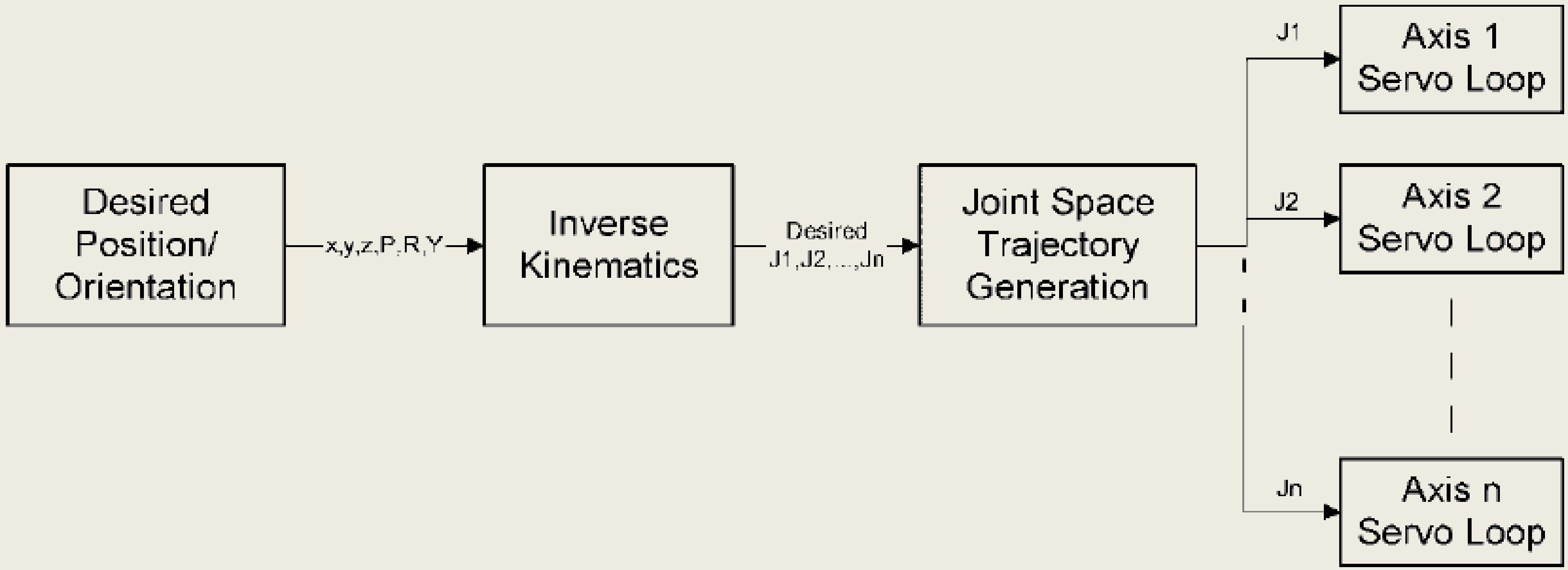
Cartesian Space Trajectories

Straight line motion can be produced if the trajectory generation is carried out in Cartesian space:

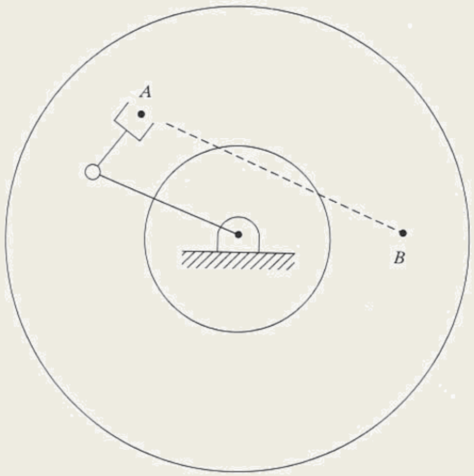
1. The trajectory is applied to the linear coordinates and angles which represent the end-effector location ${}^0T_6$.
2. These trajectories are then sampled to produce a set of ${}^0T_6$ transformations spanning the whole movement.
3. Performing the inverse kinematics calculation on each one of these gives a set of joint angle vectors for the manipulator to follow.

Trajectory Generation

Each end-effector position calculated using IK.

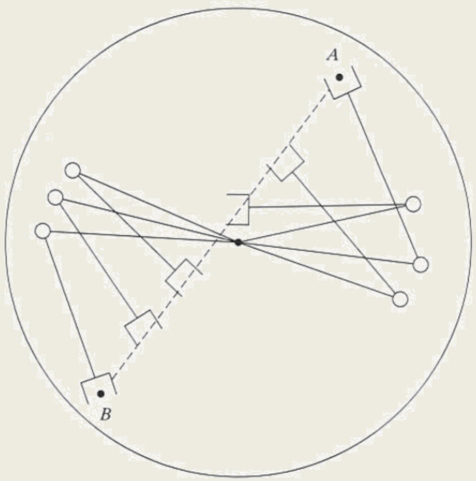


Cartesian Space Difficulties



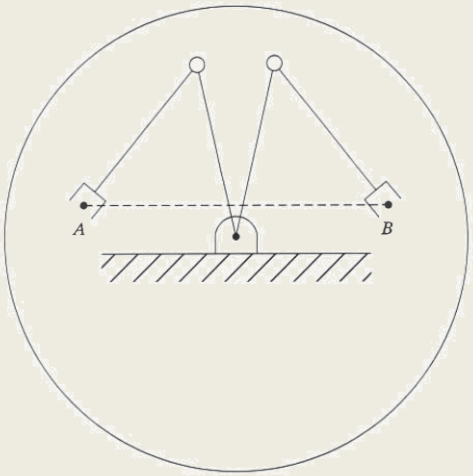
Type A

Points not in workspace



Type B

Path through singularity



Type C

Different configuration

Cartesian Space -Trajectory Constraints

Spatial – obstacles

Temporal – timing issues

Smoothness – avoiding jerky movements

Cartesian Space -Trajectory Generation

Goal:

Turn a specified Cartesian space trajectory of P_e into appropriate joint position reference values

Steps:

Use inverse kinematics of a robot manipulator arm to find joint values for any particular location of P_e

Use sampling and curve fitting to reduce computation

Output:

A series of joint position/velocity reference values to send to the controller

Cartesian Space -Trajectory Generation (2)

Step	Mode
Obtain function for path	C
Sample function to get discrete joint points	D
Apply IK & Jacobian calculations	D
Fit function to joint points	C
Sample to get discrete reference points	D

C=continuous D=discrete

Example : Linear Motion

Express line as continuous function (i.e. parameterise by time):

$$x(t), y(t)$$

Suppose we specify the line $y = mx + b$

Want to move along line with constant speed, u

Example : Linear Motion
Parameterise the line

Equation of line $$y = mx + b$$

Differentiate $$\dot{y} = m\dot{x}$$

Planar Velocity vector $$\vec{u} = \dot{x}\hat{i} + \dot{y}\hat{j}$$

Substitute $\dot{y}$ $$\vec{u} = \dot{x}\hat{i} + m\dot{x}\hat{j}$$

Example : Linear Motion

Parameterise the line (2)

$$\vec{u} = \dot{x}\hat{i} + m\dot{x}\hat{j}$$

Velocity (magnitude)

Example : Linear Motion
Parameterise the line (2)

$$\vec{u} = \dot{x}\hat{i} + m\dot{x}\hat{j}$$

Velocity (magnitude)

$$u = \sqrt{\dot{x}^2 + (m\dot{x})^2} = \pm\dot{x}\sqrt{1+m^2}$$

Solve for the *x* element (pick appropriate sign)

$$\dot{x} = \frac{\pm u}{\sqrt{1+m^2}}$$

Example : Linear Motion

Parameterise the line (3)

$$\dot{x} = \frac{\pm u}{\sqrt{1+m^2}}$$

And by substituting and then integrate we get:

$$\dot{x} = \frac{u}{\sqrt{1+m^2}} \qquad x(t) = \frac{ut}{\sqrt{1+m^2}} + x_0$$

$$\dot{y} = \frac{um}{\sqrt{1+m^2}} \qquad y(t) = \frac{umt}{\sqrt{1+m^2}} + y_0$$

Example : Linear Motion

Sample the continuous Path Function

Use M samples of the total time t_{total}

$$t_i = \left(\frac{t_{total}}{M-1} \right) i, i = 0, \ldots, M$$

$$x_i = \frac{u t_i}{\sqrt{1+m^2}} + x_0$$

$$y_i = \frac{u m t_i}{\sqrt{1+m^2}} + y_0$$

Trajectory Generation
Numerical Example : Linear Motion

Let's take a 2-link planar arm with

- Link lengths: $l_1 = 4$, $l_2 = 3$ m
- Start point = $(0, l_1 + l_2 /4)=(0,4.75)$
- End point = $(l_1 + l_2 /4,0)=(4.75,0)$
- Constant speed $u=3\ m/s$

Note, in practice, speed usually follows a trapezoidal profile with acceleration/deceleration at start/end of the motion

Numerical Example : Linear Motion Step 1 – Establish the function for path

The path has the equation:

$$y = -x + l_1 + l_2 / 4 \Rightarrow \boxed{y = -x + 4.75}$$

And a length of:

$$dist = \sqrt{\left(l_1 + \frac{l_2}{4}\right)^2 + \left(l_1 + \frac{l_2}{4}\right)^2} \approx 6.72m$$

Which for a constant speed will take:

$$t_{total} = \frac{dist}{u} = \frac{6.72}{3} \approx 2.24s$$

Numerical Example : Linear Motion Step 2 – Sample the Function

We will use M=9 sample points

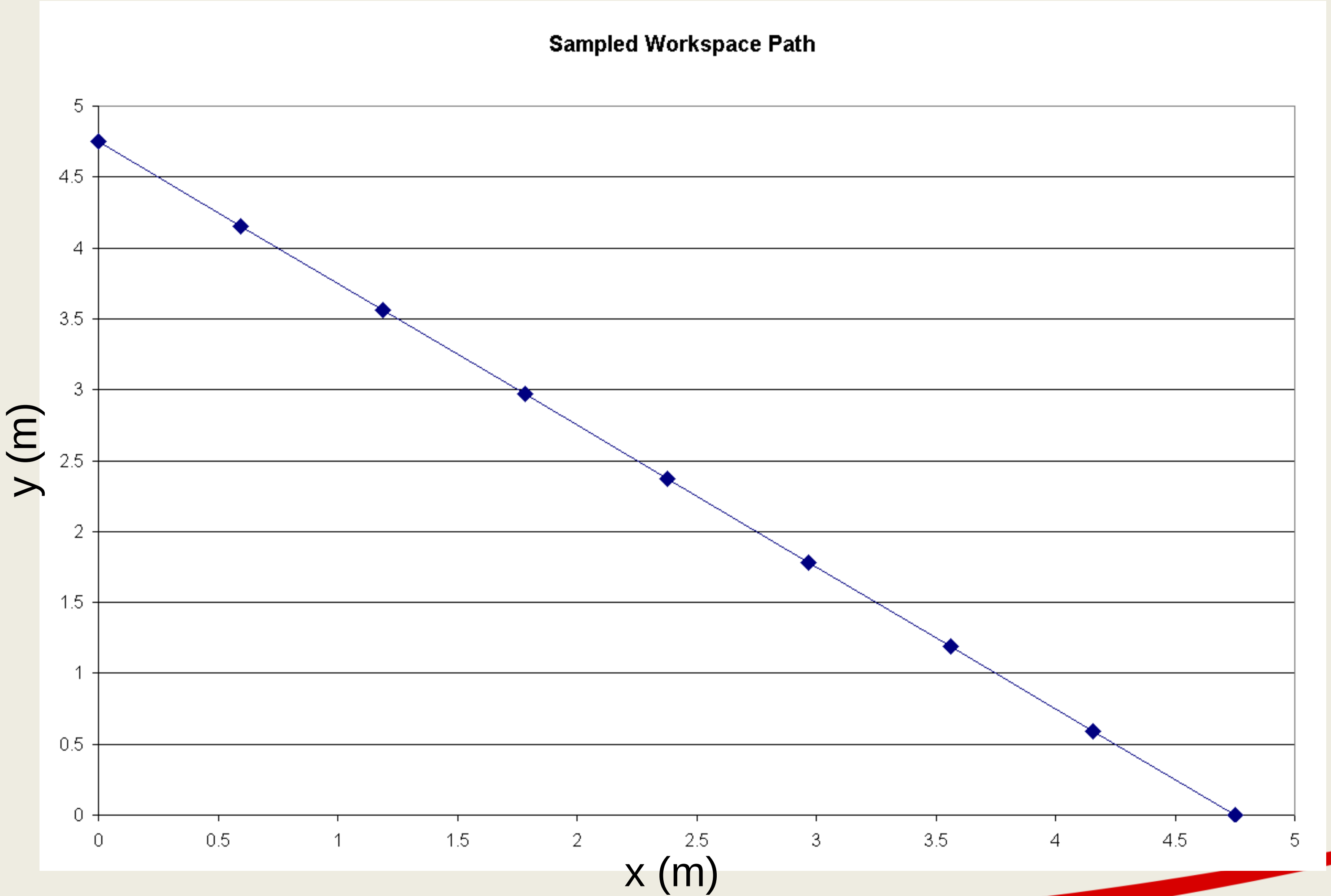
$$t_i = \left(\frac{t_{total}}{M-1} \right) i = \left(\frac{2.24}{9-1} \right) = 0.28i, i = 0, \ldots, 9$$

And for constant speed we will have

$$x_i = \frac{ut_i}{\sqrt{1+m^2}} + x_0 = \frac{3t_i}{\sqrt{2}}$$

$$y_i = \frac{umt_i}{\sqrt{1+m^2}} + y_0 = \frac{-3t_i}{\sqrt{2}} + 4.75$$

Numerical Example : Linear Motion Step 2 – Sample the Function



Numerical Example : Linear Motion Step 3 – IK on the Sampled points

Position Inverse Kinematics:

$$\theta_2 = a\tan 2\left[\sin\theta_2, \cos\theta_2\right]$$

$$\theta_1 = a\tan 2\left[y, x\right] - a\tan 2\left[l_2 \sin\theta_2, l_1 + l_2\cos\theta_2\right]$$

where

$$\cos\theta_2 = \left(x_2 + y_2 - l_1^2 - l_2^2\right)/\left(2 l_1 l_2\right)$$

$$\sin\theta_2 = \pm\sqrt{1 - \cos^2\theta_2}$$

Numerical Example : Linear Motion Step 3 – IK on the Sampled points

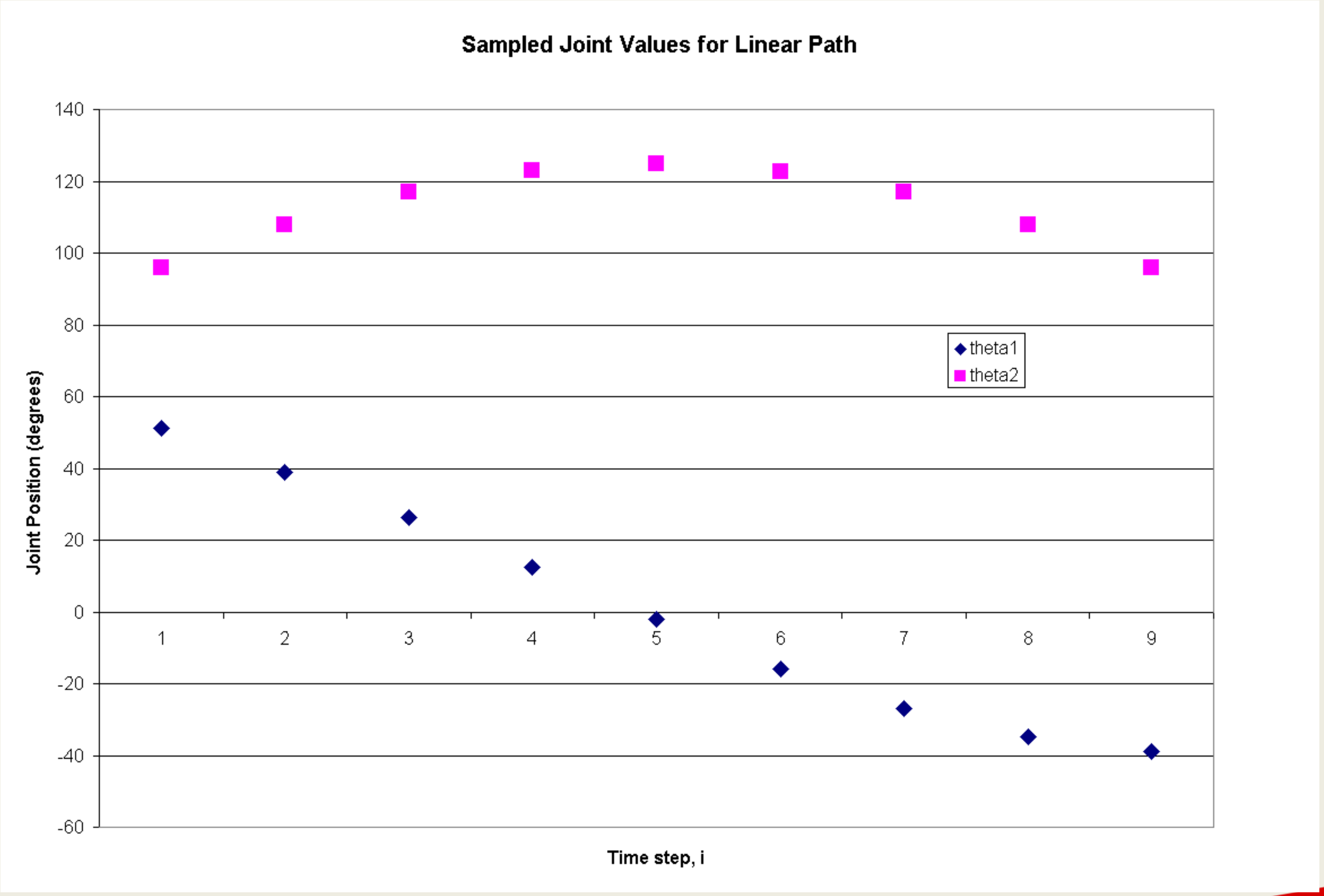
Velocity Inverse Kinematics:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = [J] \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

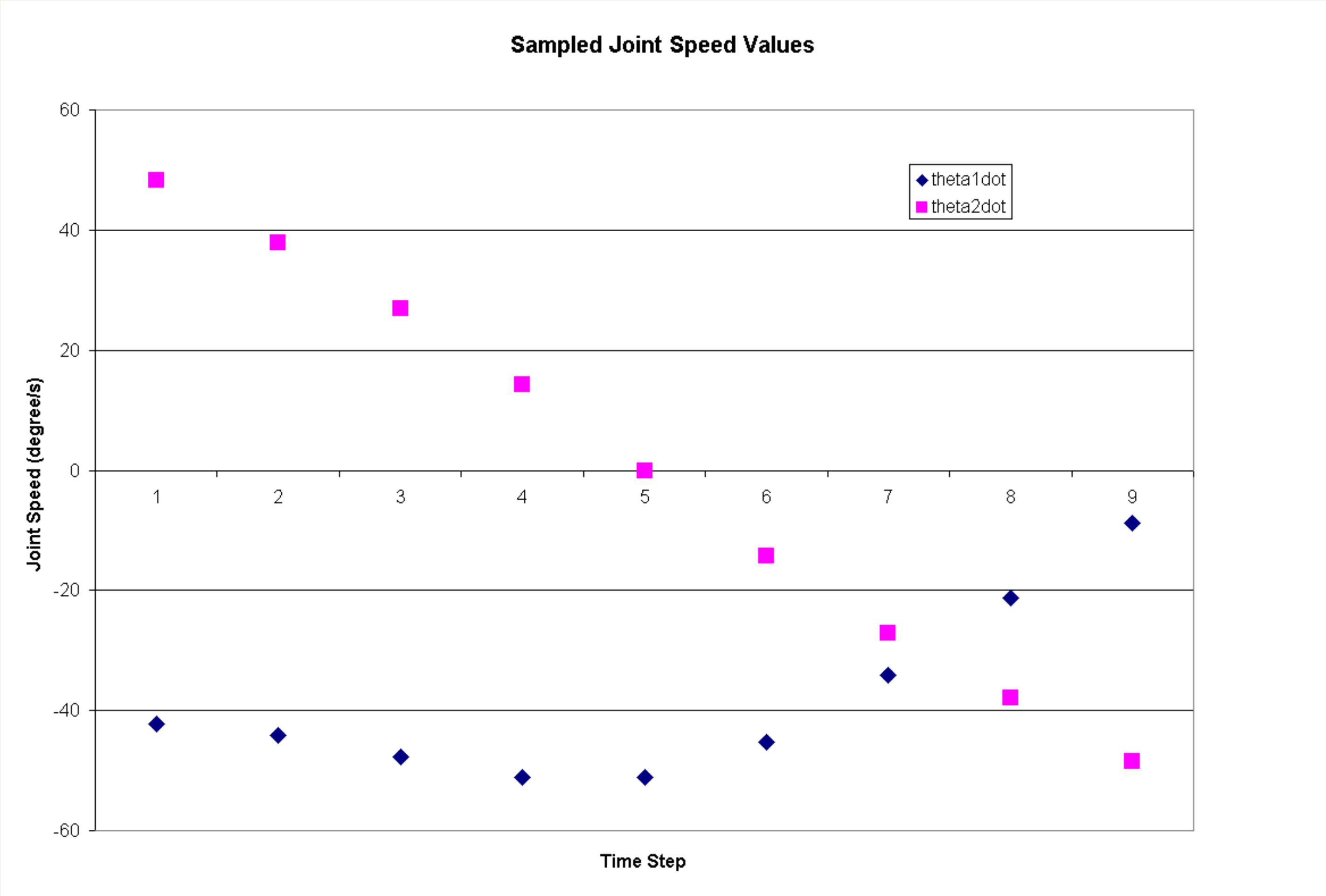
where

$$J = \begin{bmatrix} -l_1 \sin\theta_1 - l_2 \sin(\theta_1 + \theta_2) & -l_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \end{bmatrix}$$

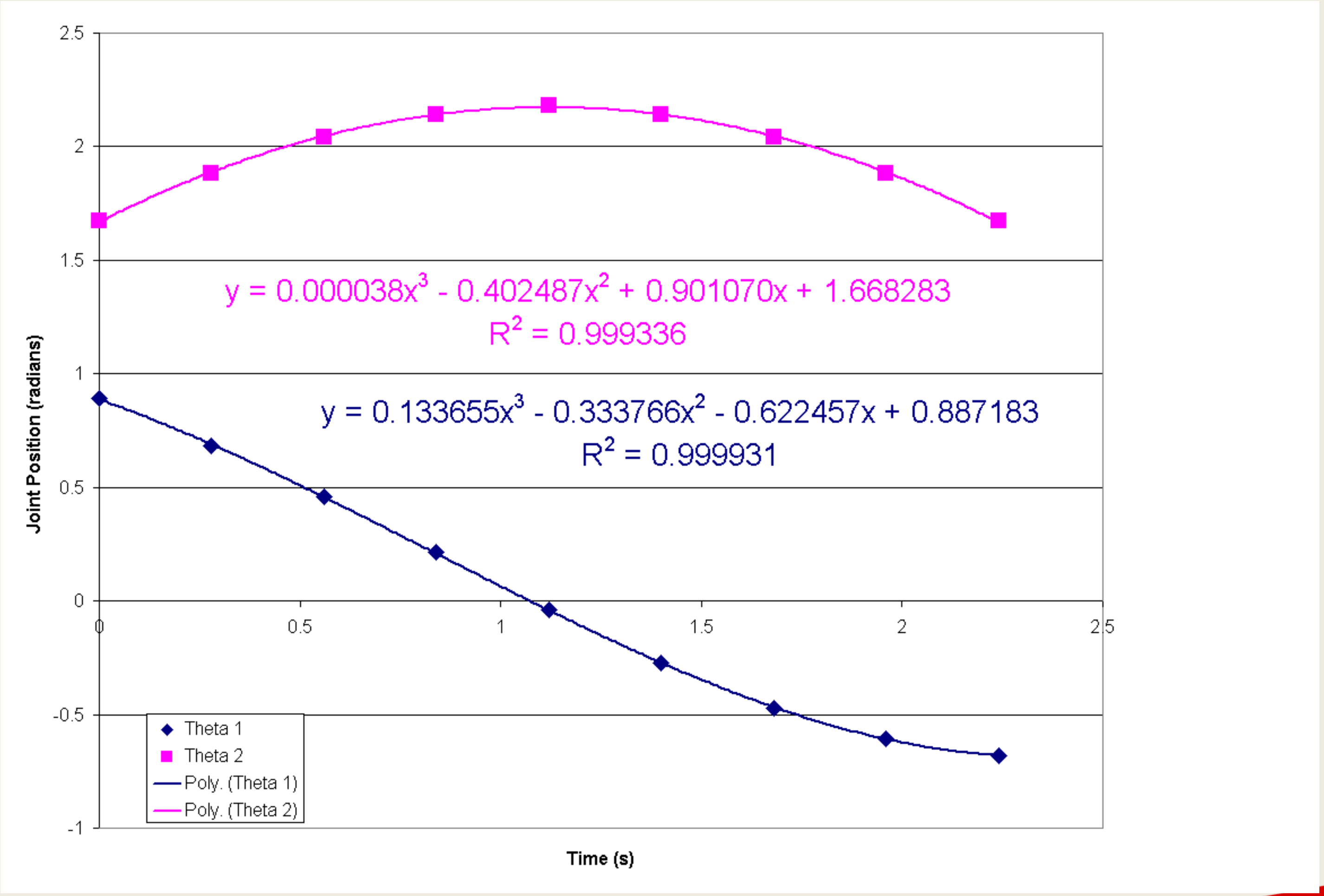
Numerical Example : Linear Motion Step 3 – Joint Position Values from IK



Numerical Example : Linear Motion Step 3 – Joint Position Values from IK



Numerical Example : Linear Motion Step 4 – Fit Cubic to Position/Velocity

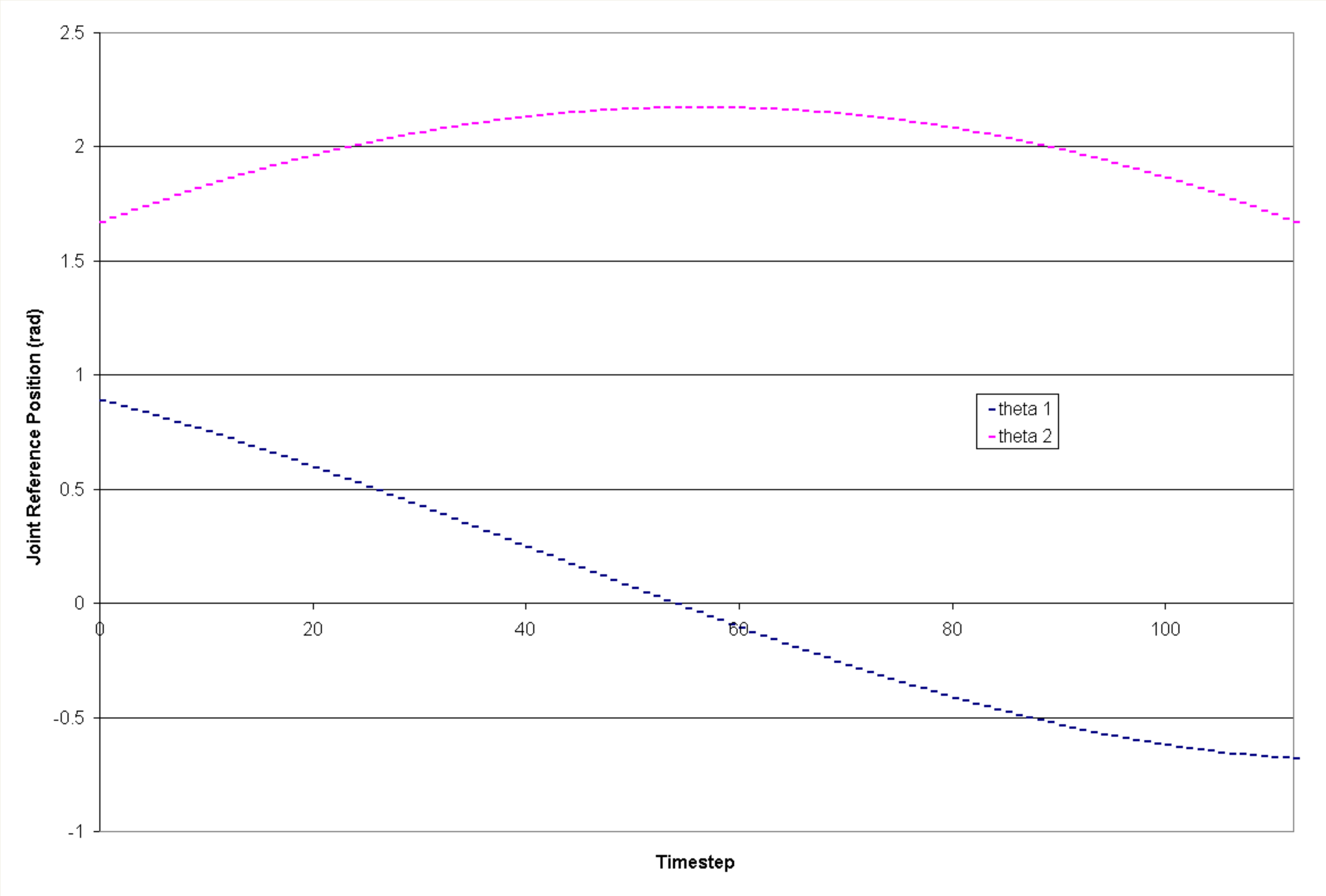


Numerical Example : Linear Motion Curve Fitting Comments (Step 4)

Typically, a single cubic is not sufficient for the entire motion

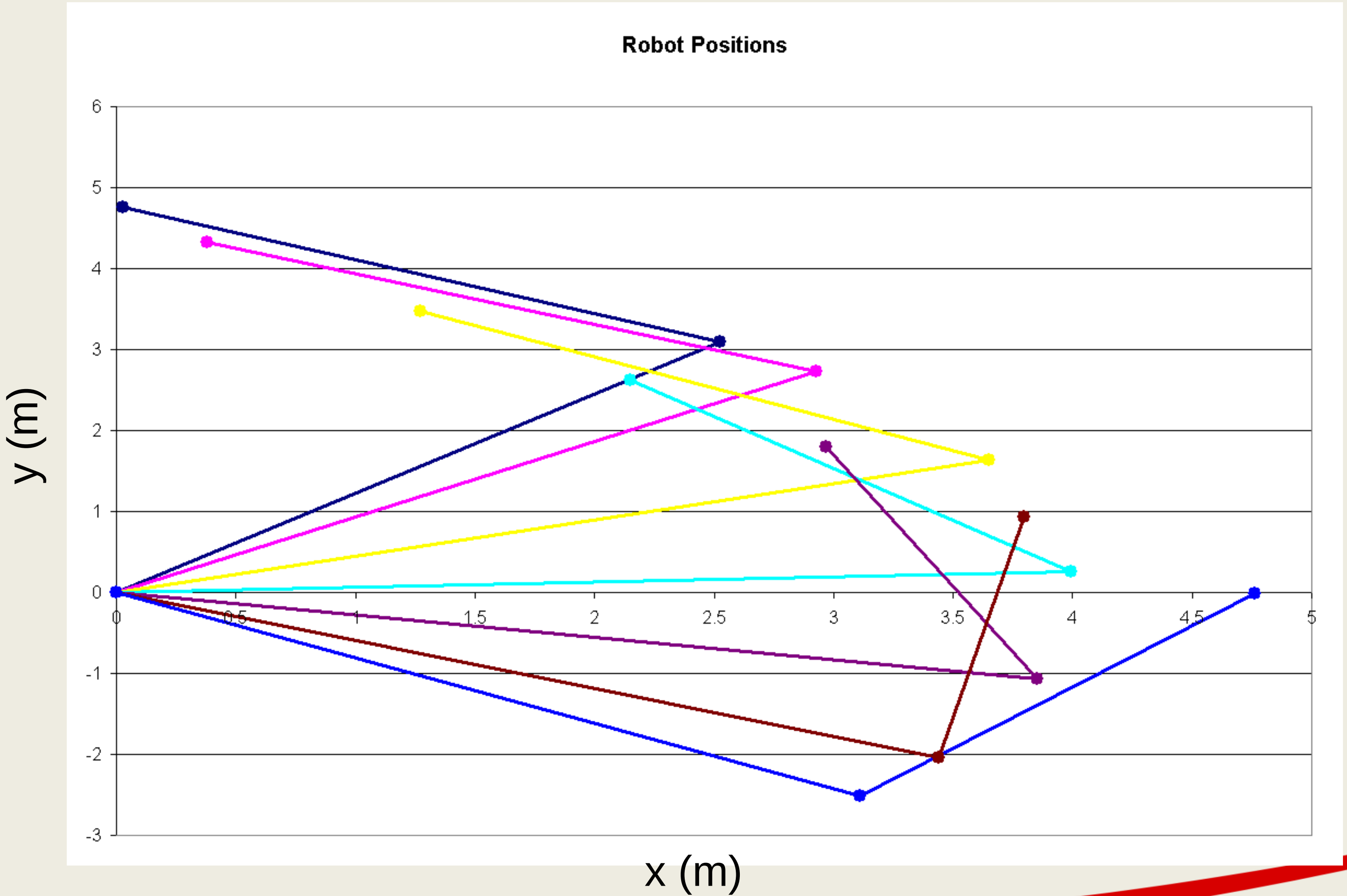
A quintic polynomial can fit position, velocity and accelerations of end-points

Numerical Example : Linear Motion Step 5 – Sample Equation to get points



Sample at 0.02s

Numerical Example : Linear Motion Trajectory Followed



Run time path generation

Joint-space paths

Coefficients fed to the control system at the end of each segment (high order poly) or checked if in linear or blend portion (linear with blends)

Cartesian space paths

Use the linear with blends path generator but use x, y instead of the joint angles in the equations. These are then converted in the joint angles using IK.

Collision-free path planning

Local vs global motion planning

- Gross motion planning for uncluttered areas
- Fine motion planning for the end effector frame

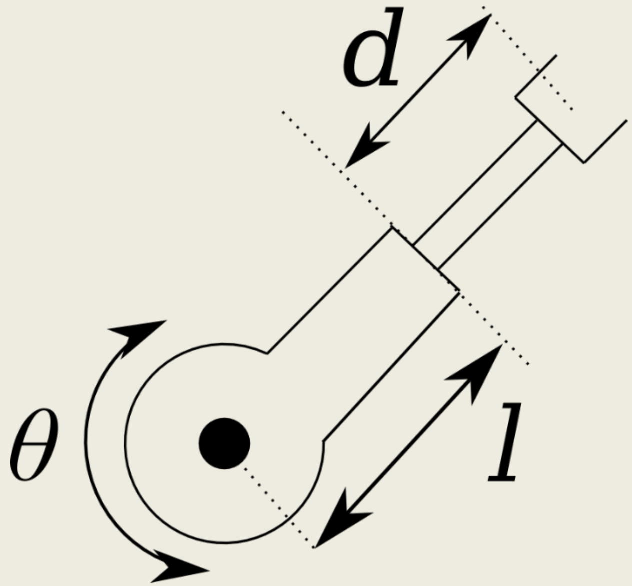
Configurations space approach

Artificial potential field approach

Example

Consider the robot of the figure below. You are required to generate two linear trajectories in the Cartesian Space (x, y):

- From (1, 1) to point (-0.5, 1.5)
- From (1, 1) to point (0.5, -1)



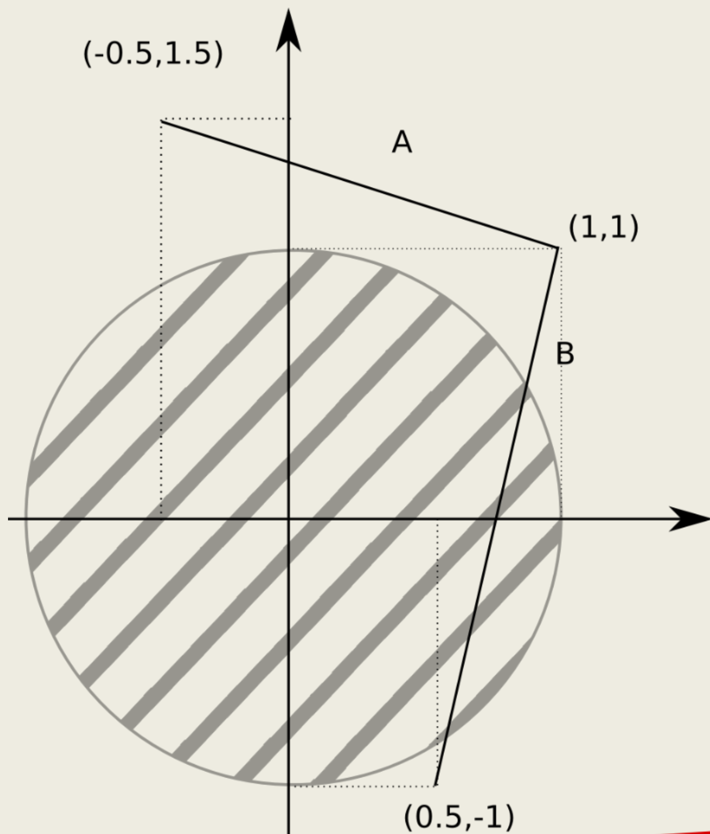
The characteristics of the robot are:

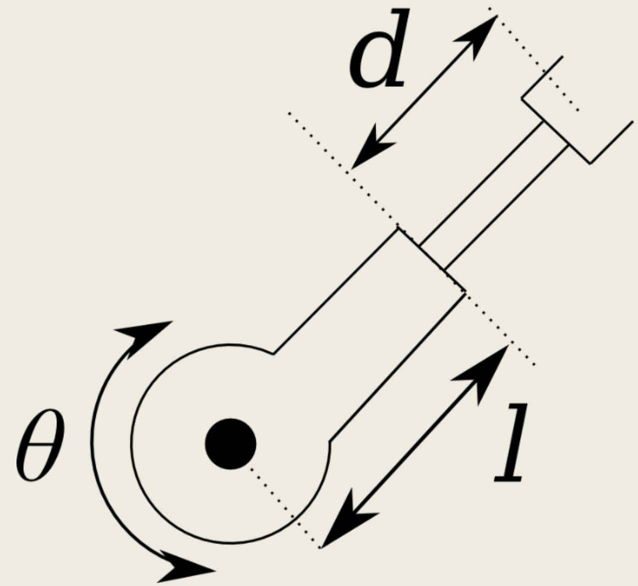
- l = 1m
- θ can take the values from 0º to 360º
- d can take the values from 0m to 1m

Can you achieve both trajectories?
If not, what is the problem? Draw it.

Example

- From (1, 1) to point (-0.5, 1.5)
- From (1, 1) to point (0.5, -1)
- l = 1m
- θ can take the values from 0º to 360º
- d can take the values from 0m to 1m





Conclusions

Via points have different calculations for different parts of the trajectory

Linear motion achieved only in Cartesian Space

Cartesian Space might lead manipulator to singular configurations

Cartesian Space calculations are happening in discrete and continuous methods

Exercise:

You are required to create a trajectory for a joint to move between two points (θ_1 = 0^o and θ_4 = 30^o) via two other points (θ_2 = 40^o and θ_3 = 20^o). Your requirements are, that all blend times must be 2sec and all durations between points should be 5sec. Finally, you are also required to use the motor in stock that can produce an acceleration of magnitude of 6.

a) Can you satisfy all requirements of the trajectory?
b) If not, what amendments in the requirements you can make? What are the
new characteristics of the trajectory?
c) Plot/draw the velocity and acceleration profiles of the movement