# ROBOTIC SYSTEMS (UMFM4X-15-M)

Trajectories

UWE Bristol
University of the West of England

# Previously on ROBOTIC FUNDAMENTALS

**Parallel Robot kinematics:**

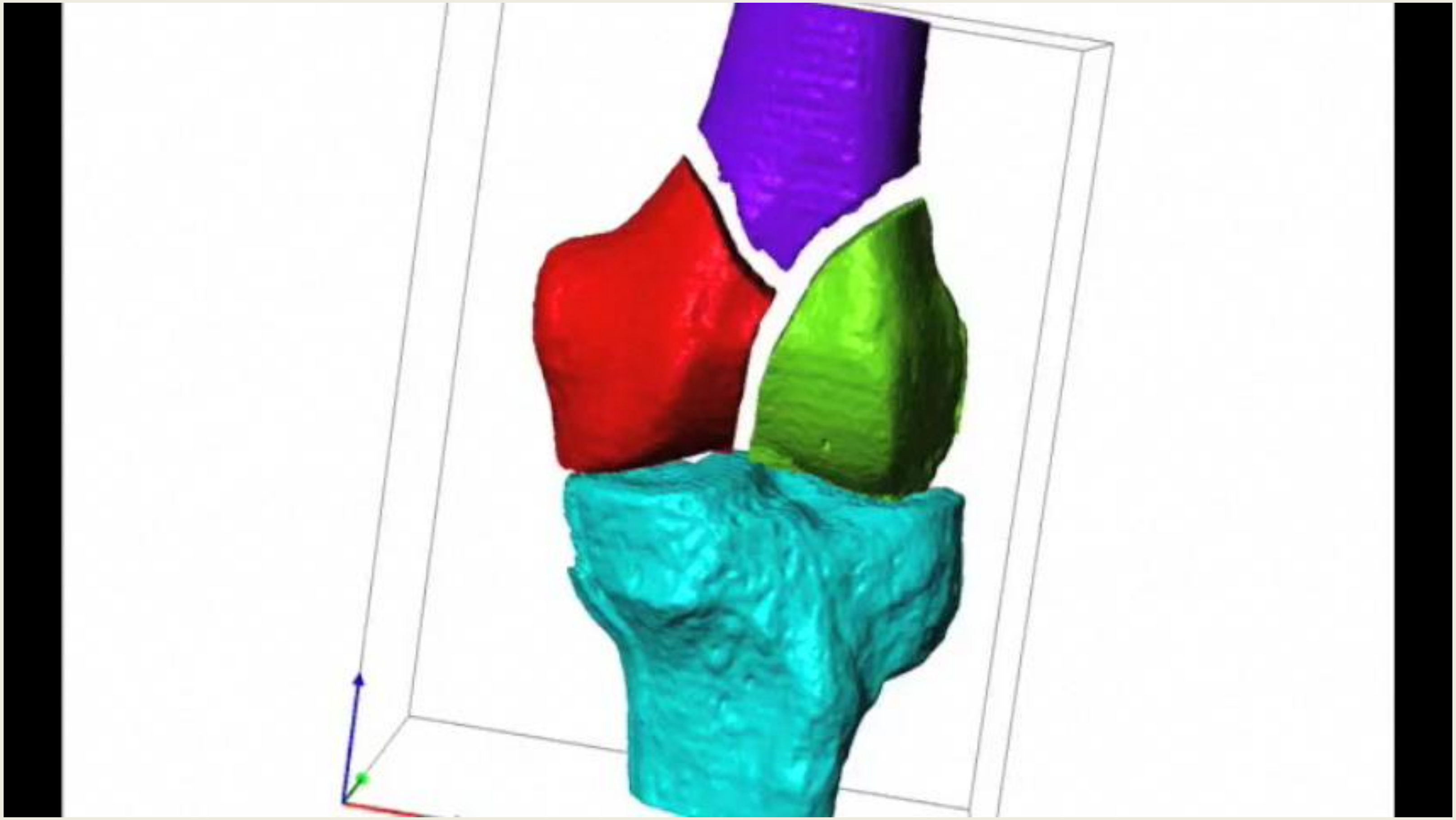
- The workspace of Parallel robots can be calculated geometrically or numerically.
- IK is performed first by vector chain and then by defining magnitude and attitude of vector
- FK of parallel robots are computationally complicated

Questions?

# Today’s lecture

Path Selection

Joint Space

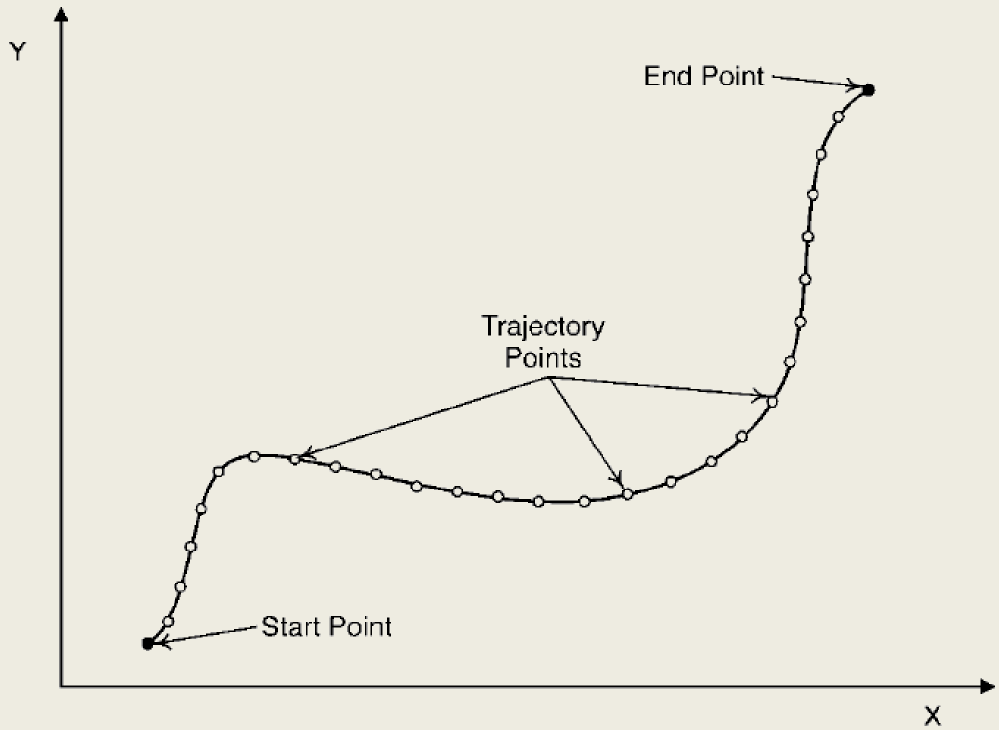


# Outline

- Path Selection
- Path Generation – Joint Space
  - Polynomial Functions
  - Linear Functions
- Continuous Path Mode

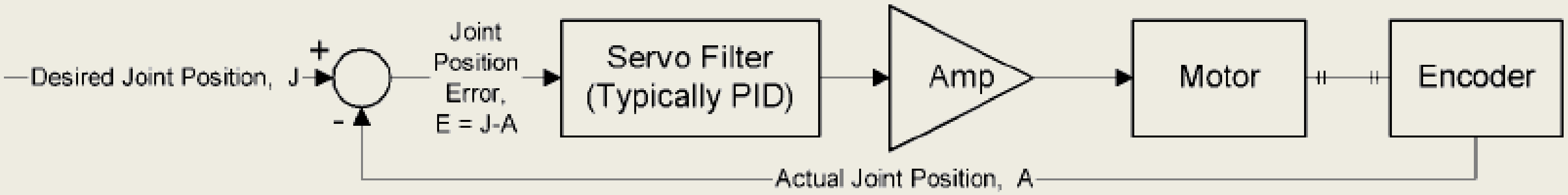
# Path Selection

**Trajectory generator** calculates the next intermediate point to be sent to the servo loop.



# Trajectory generation

Servo-driven devices with a closed loop control

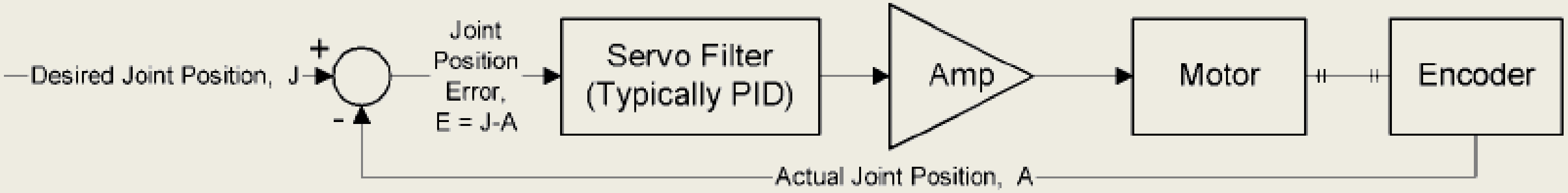



Can we control only position?

NO

# Trajectory generation

Servo-driven devices with a closed loop control



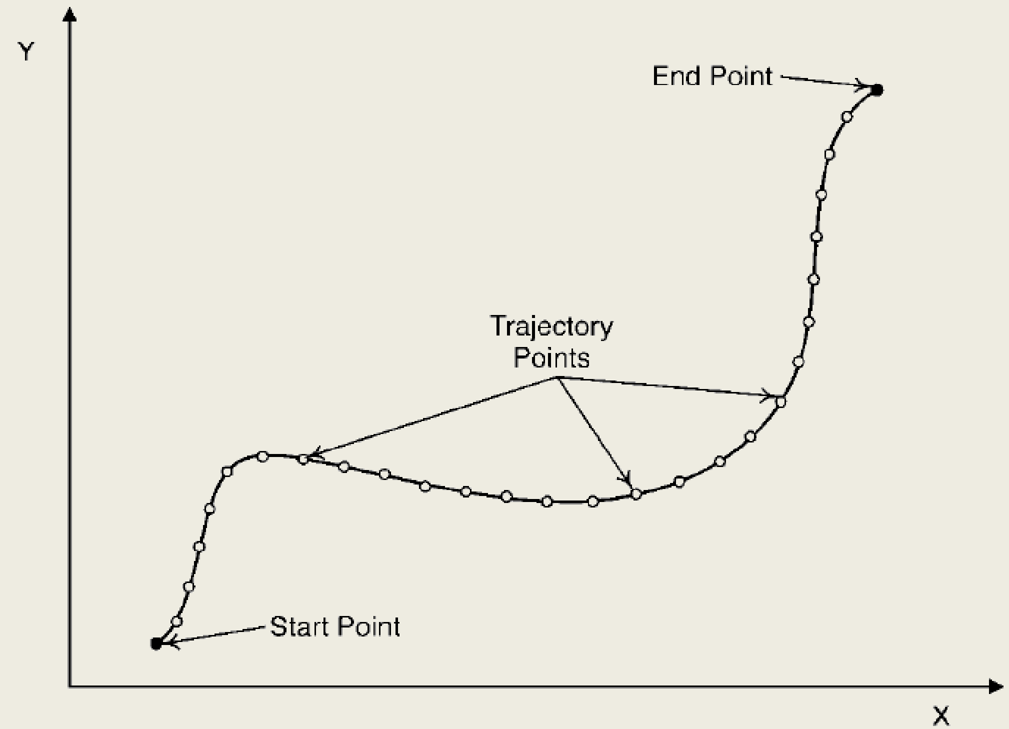

**Accelerations** and **speed** can be controlled by the rate at which points are sent to the servo loop.

The result defines a **velocity profile for the move**.

# Path Selection (2)

**Trajectory generator** calculates the next intermediate point to be sent to the servo loop.

It is calculated based upon the desired velocity profile and path.



# Path Selection (3)

Stored either as **joint values** (Joint space) or **end-effector's position and orientation** (Cartesian space)

**Joint space**: <u>no kinematic calculations are required</u>; points taught to the system by <u>jogging the robot</u> to the desired position and <u>recording the joint values</u>.

**Cartesian space**: kinematic <u>calculations are required</u> to provide these joint angles for a given location and orientation.

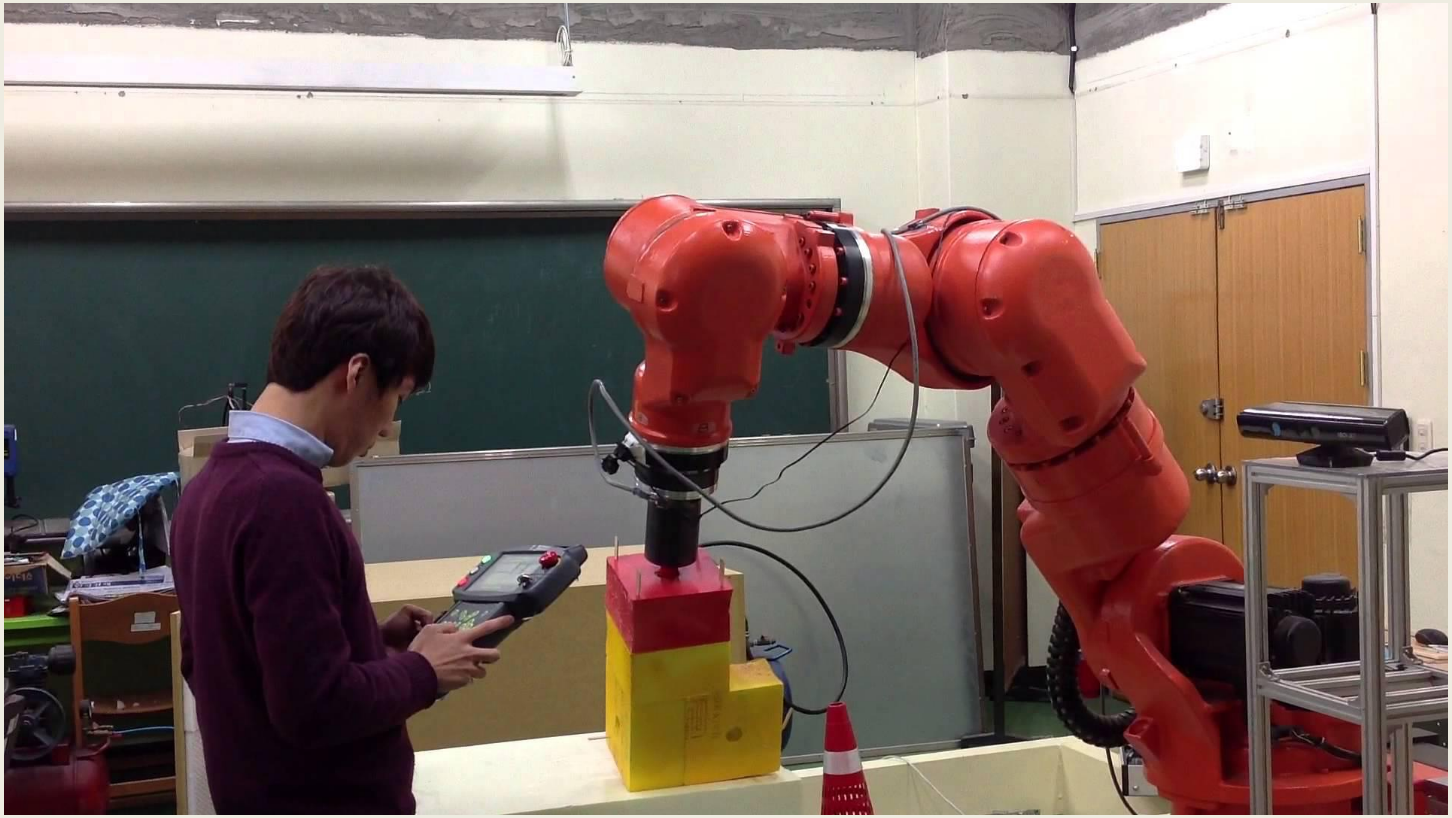

E2118

# Joint Space Trajectories

An operator manually jogs the robot to a desired position and then **stores that position** e.g. stores the joint values for that position. Once that position is taught, it can be recalled during robot operation.

<u>No need for inverse kinematic</u> calculations. The path of the end-effector between points is usually not a straight line.

# Joint Space Trajectories



Point-to-point trajectory (PPT), based upon taking the **joint values (angles) as input**. These joint values obtained through the use of a **teach pendant**.

# Joint Space Trajectories (2)


Put the robot in **gravity compensation** and move it to the desired position.

https://www.youtube.com/watch?v=8vycqIIheKk

# Joint Space Trajectories (3)

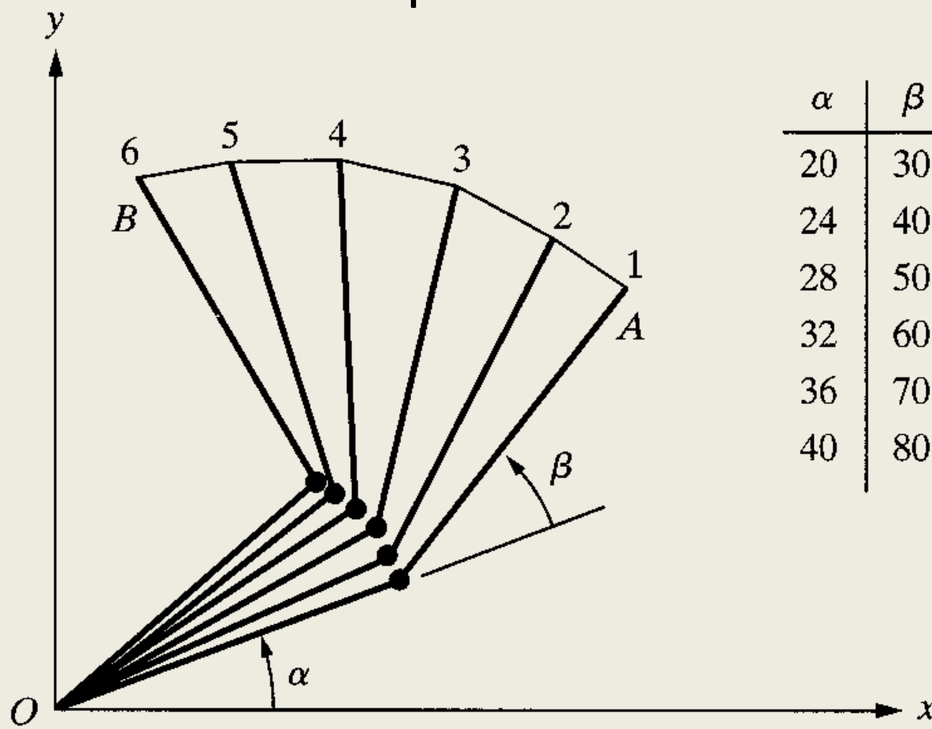
An operator manually jogs the robot to a desired position and then **store that position** e.g. stores the joint values for that position. Once that position is taught, it can be recalled during robot operation.

No need for inverse kinematic calculations. The path of the end-effector between points is usually not a straight line.

# PPT – Start and Finish

Joint α: 20 -> 40

Joint β: 30 -> 80




| $\alpha$ | $\beta$ |
|---|---|
| 20 | 30 |
| 24 | 40 |
| 28 | 50 |
| 32 | 60 |
| 36 | 70 |
| 40 | 80 |

Joint-space, normalized movements of a robot with two degrees of freedom.

# Time Change of Joint Values

Function that relates joints values to time (from initial position to final position)

**Polynomial**

$3^{rd}$ order (Cubic)

$5^{th}$ order (Quintic)

**Linear**

with Parabolic Blends

Joint Space Trajectories

# POLYNOMIAL FUNCTIONS

# Polynomial Functions

$$\theta = c_0 + c_1 t + c_2 t^2 + c_3 t^3$$

$$\theta = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5$$

The original, first derivative (velocity) and second derivative (acceleration) equations form a **system of equations** (3 & 5) with (3 & 5) <u>respective unknowns</u>

# Polynomial Function – Cubic

$$\theta = c_0 + c_1 t + c_2 t^2 + c_3 t^3$$

We set the start and end conditions for <u>position</u> and <u>velocity</u> to get the polynomial parameters

# Polynomial Function – Cubic (2)

$$\theta = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

We set the start and end conditions for position and velocity to get the polynomial parameters

By taking the derivative:

$$\dot{\theta} = a_1 + 2a_2 t + 3a_3 t^2$$

( $\ddot{\theta} = 2a_2 + 6a_3 t$ acceleration)

# Polynomial Function – Quintic

If more constraints have to be satisfied (start and end accelerations) then a **fifth order** polynomial can be used

$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5$$

# Polynomial Function – Quintic (2)

If more constraints have to be satisfied (start and end accelerations) then a **fifth order** polynomial can be used

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5$$

With the derivatives:

$$\dot{\theta}(t) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4$$

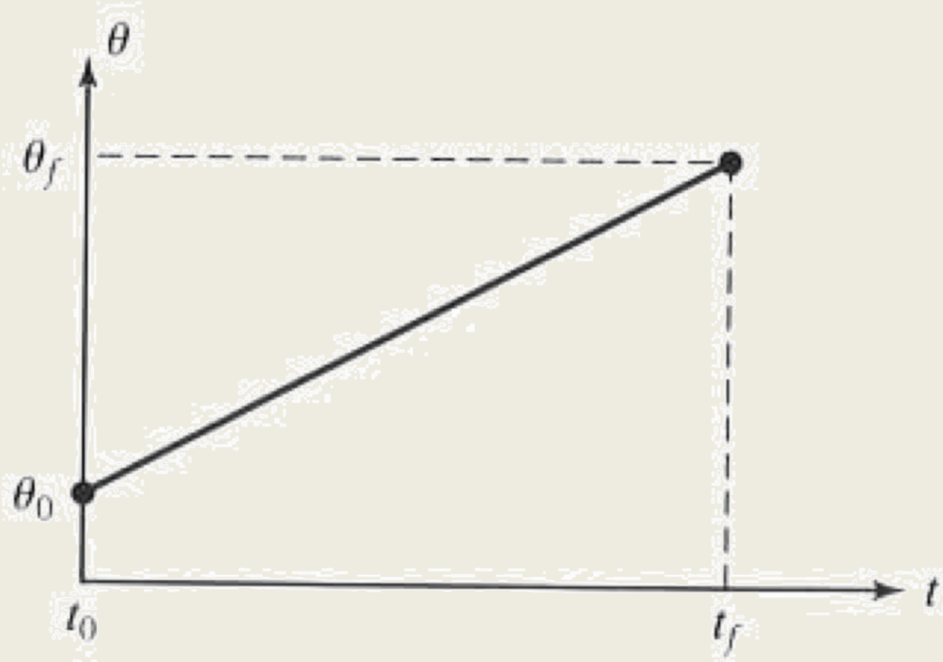
$$\ddot{\theta}(t) = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3$$

Joint Space Trajectories

# LINEAR FUNCTIONS

# Linear functions

Linear functions can also be fitted between the joint values



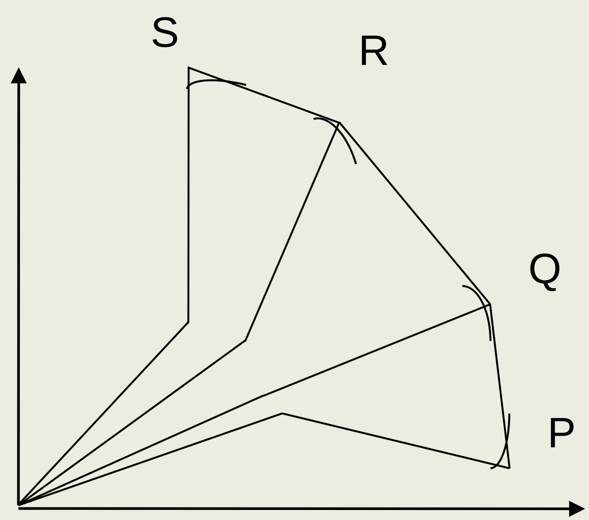
# Linear functions – Parabolic Blends

Linear functions can also be fitted between the joint values

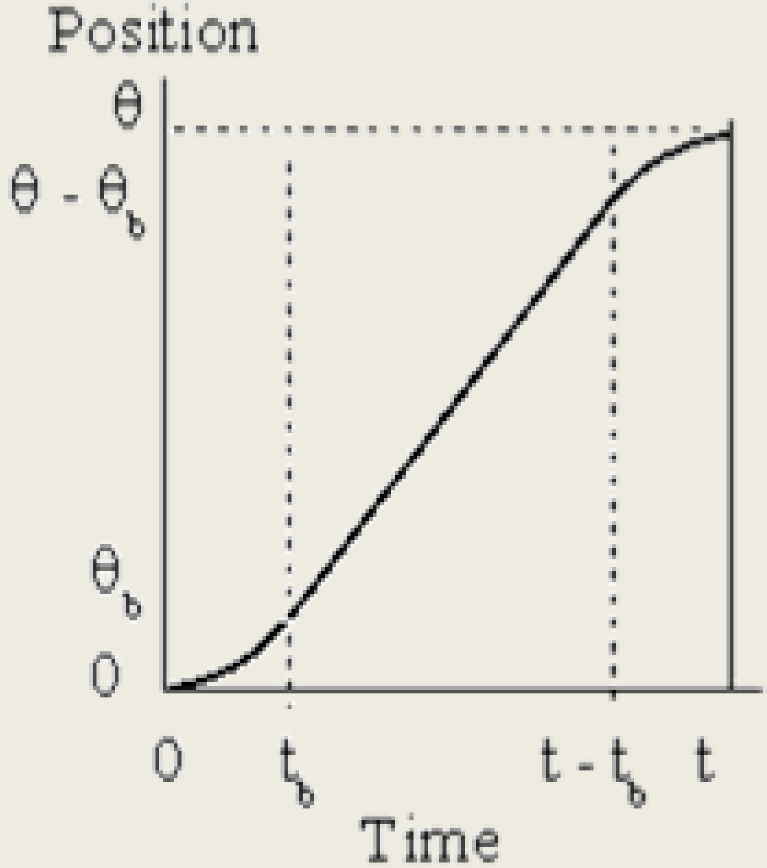
**BUT**

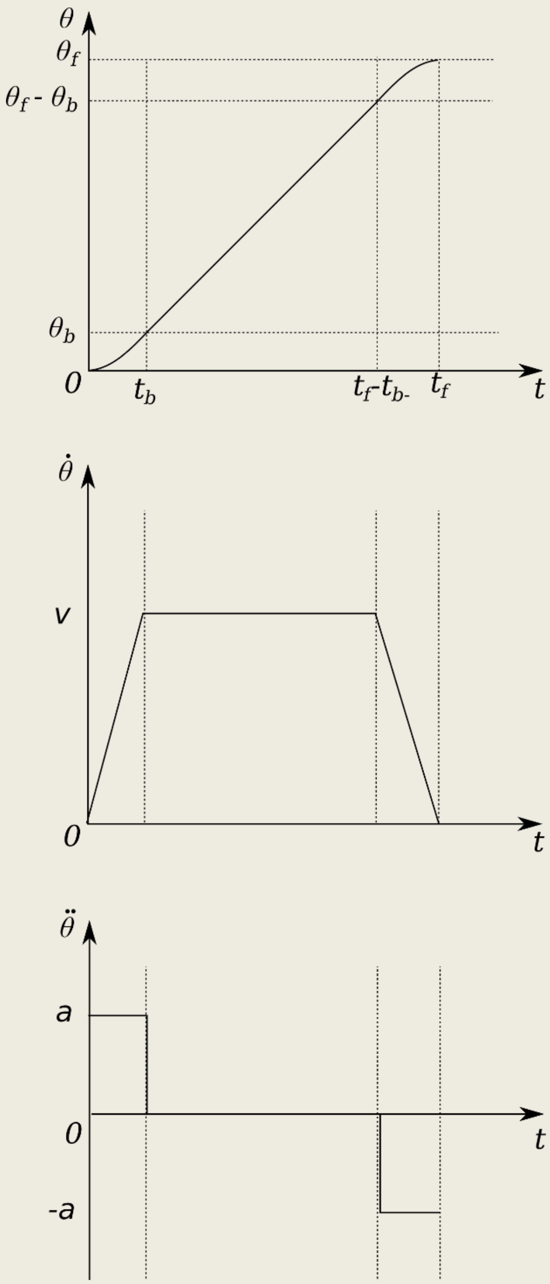
Velocity to **be discontinuous** at the beginning and at the end

To create a smooth function for path we add **parabolic blends**



# Linear functions – Parabolic Blends






# Linear function with parabolic blends

$V = \dfrac{\theta_e - \theta_s}{t_e - t_s}$

linear.

Remember These!

$$t_b = \frac{v}{a} \quad (1)$$

$$\theta_b = \frac{v \cdot t_b}{2} \quad (2)$$

$$\theta - \theta_b - \theta_b = v \cdot (t - t_b - t_b) \quad (3)$$

$$\theta - v \cdot t_b = v \cdot (t - 2t_b) \Rightarrow \theta = v \cdot (t - t_b)$$

$$\theta = v \cdot t - \frac{v^2}{a}$$

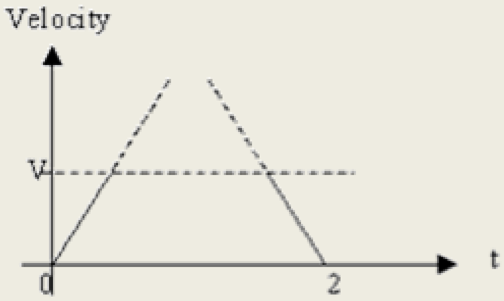
# Linear function with parabolic blends – Selecting correct value

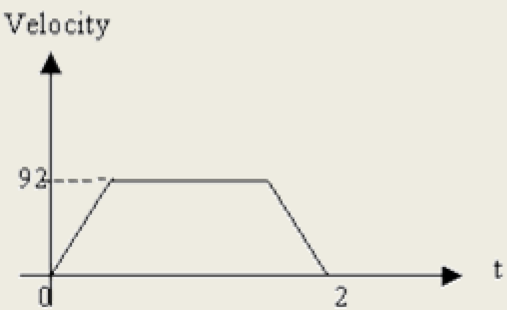
A joint moves for a total of $150^o$ in 2 seconds.

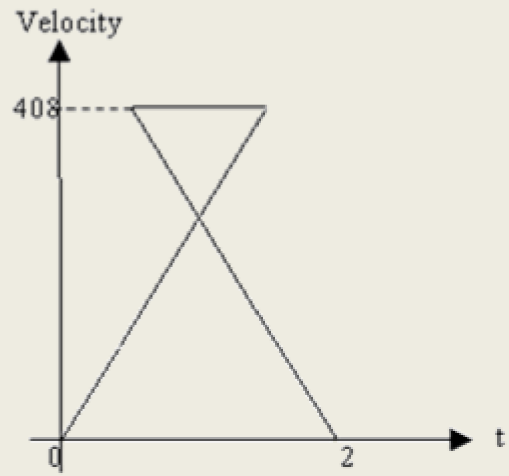
Acceleration and deceleration is at $250^o/s^2$ .

Calculate constant velocity value.

# Linear function with parabolic blends – Selecting correct value



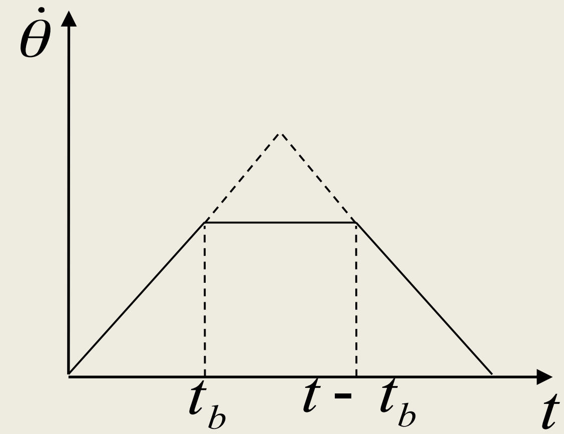




Smaller value is always the correct

# Linear function with parabolic blends – Special acceleration concern

How much acceleration can you have?

$$\ddot{q} \cdot t_b = \frac{q - q_b}{t - t_b}$$




$$q_b = \frac{v \cdot t_b}{2} = \frac{1}{2} \cdot \ddot{q} \cdot t_b^2$$

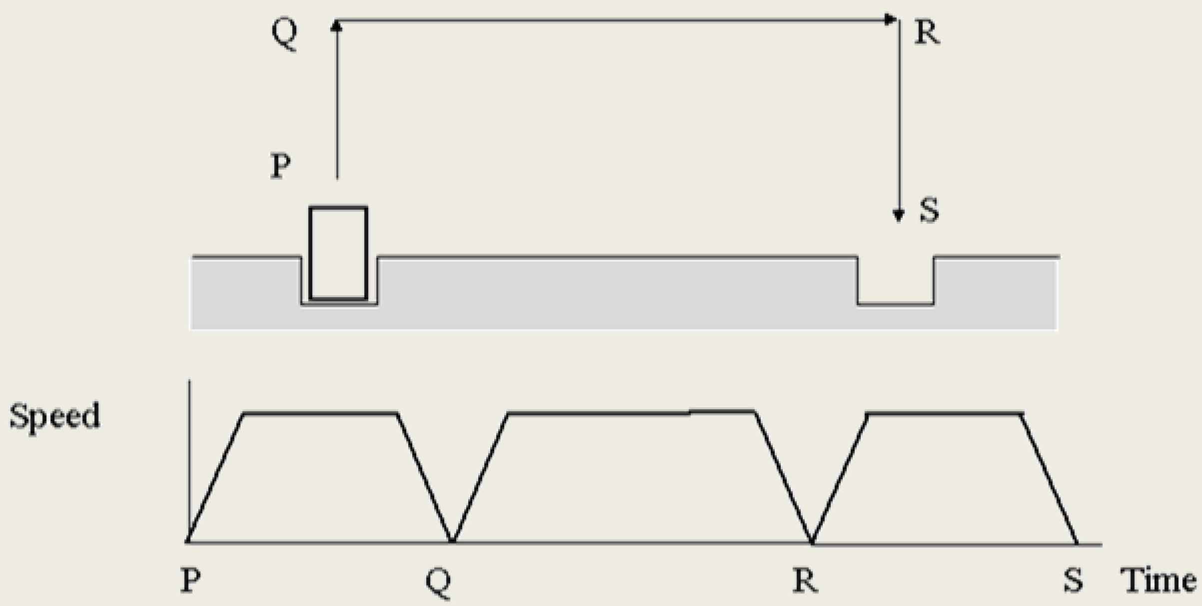
$$\ddot{q} \cdot t_b^2 - \ddot{q} \cdot t \cdot t_b + q - q_0 = 0$$

$$t_b = \frac{t}{2} - \frac{\sqrt{\ddot{q}^2 t^2 - 4 \cdot \ddot{q} \cdot (q - q_0)}}{2\ddot{q}} \qquad \ddot{q} \geq \frac{4(q - q_0)}{t^2}$$

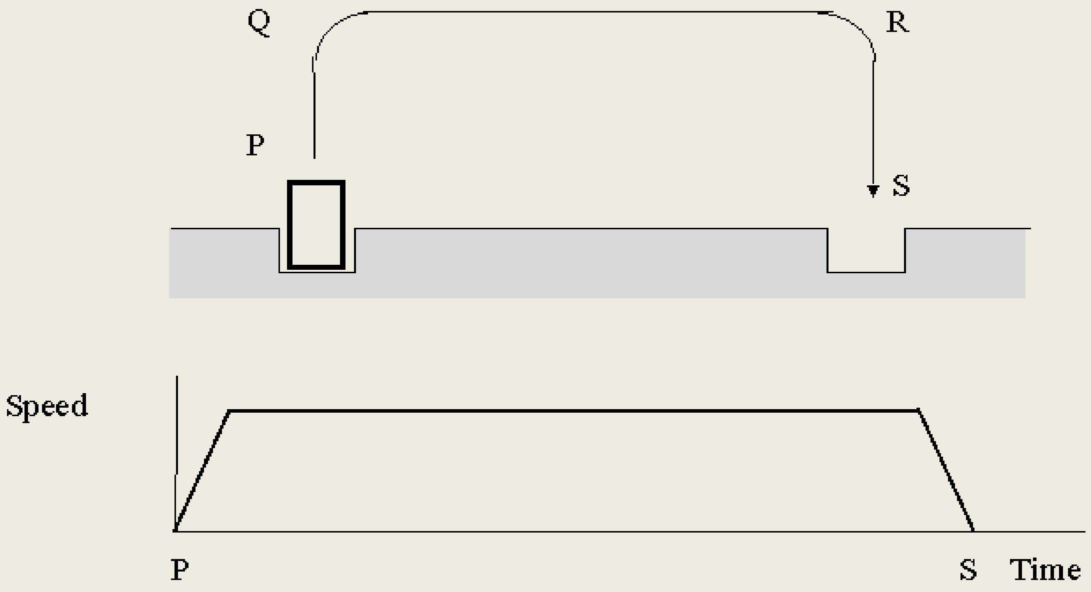
Joint Space Trajectories

# CONTINUOUS PATH MODE (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 is changes

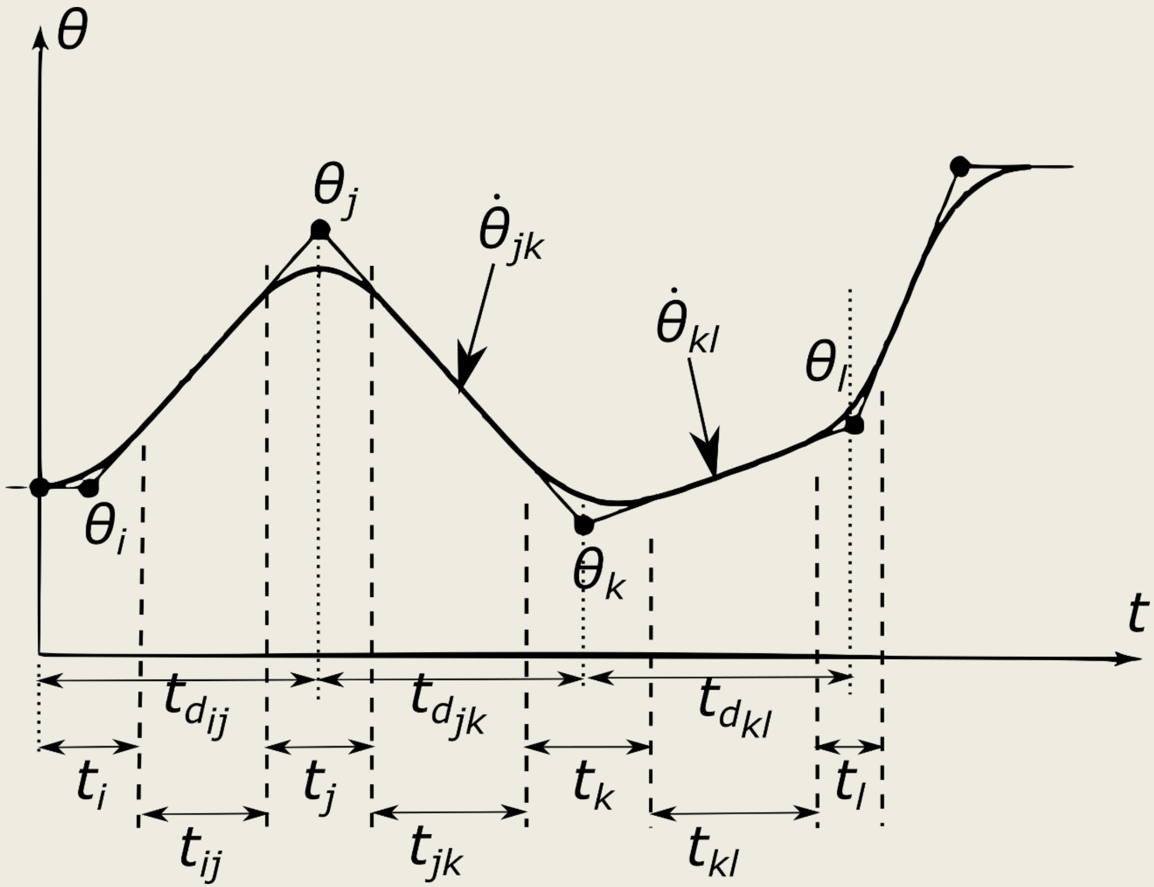
Via points described in terms of a desired position and orientation

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 (more next week)




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

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

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

# Conclusions

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.


37
37