ZENITH / STUDY HUB

SPPU | TE Robotics & Automation | Semester V

PCC303-ROA • UNIT V

TRAJECTORY PLANNING & CONTROL

Complete syllabus-aligned unit notes • 2024 Pattern

Theory • diagrams • derivations • numericals • exam answers

- Trajectory generation: joint space versus Cartesian
- Cubic, quintic, LSPB and spline trajectories
- Velocity, acceleration, torque and joint constraints
- Open / closed-loop control and PID
- Feedback linearization and computed torque
- Simulation workflow and practical implementation

Source: RKD Complete Exam Notes, Edition 3 • source pages 93–116

Extracted without modifying original teaching diagrams or formulas

AUTHOR Om Deshmukh

How to use Unit V

PCC303-ROA | SPPU TE Robotics & Automation | Semester V
2024 Pattern | 8 teaching hours | Official syllabus, printed page 20

01 STUDY PLAN

1. Start with the difference between geometric paths and time laws.
2. Solve cubic, quintic, LSPB and spline examples step by step.
3. Redraw motion-profile diagrams and controller feedback loops.
4. Check actuator and motion constraints for each trajectory.
5. Derive the computed-torque control law before memorizing it.
6. Revisit simulation examples and the Unit V question bank.

02 QUICK TOPIC FINDER

Original source pages	Topics
93–97	Trajectory generation; joint vs Cartesian space
97–104	Cubic, quintic, LSPB, via points and splines
104–106	Trajectory constraints and numerical examples
106–110	Control architectures and PID illustrations
110–112	Feedback linearization and computed torque
113–116	Simulation, practical aspects and question bank

SOURCE AND STUDY NOTES

Edition 3 teaching material: original physical PDF pages 93–116.
Original diagrams and equations are retained in this review edition.
Cubic/quintic boundary-condition checks are documented in the QA work.
Suggested exam-answer lengths are advice, not guaranteed paper structure.

UNIT V · 08 TEACHING HOURS

Unit V · Trajectory Planning and Control

V

SYLLABUS TOPICS IN THIS UNIT

- 5.1 Trajectory generation
- 5.2 Polynomial & spline trajectories
- 5.3 Trajectory constraints
- 5.4 Control architectures
- 5.5 Feedback linearization
- 5.6 Simulation & practical implementation

AFTER THIS UNIT YOU CAN

Plan smooth joint motions (cubic, quintic, LSPB, via points), respect speed and torque limits, and choose a controller (PD, PID, computed torque) for a robot arm (CO5).

TYPICAL QUESTION TYPES (STUDY GUIDE, NOT AN OFFICIAL PATTERN)

Theory (6 marks): joint vs Cartesian space, why smooth paths, control loops, PID terms, computed torque / feedback linearization. Numericals (7 marks): cubic or quintic coefficients and peak velocity, LSPB blend time, PD gains.

IN SIMPLE WORDS

Units II–IV said where the robot can go and what torque it needs. Unit V decides how it should move over time (trajectory) and how to make it actually follow that plan (control).

SYLLABUS TOPIC

5.1 Trajectory Generation

IN SIMPLE WORDS

A path is just the route (a set of points). A trajectory is the route plus timing: where the robot must be at every instant, with its speed and acceleration.

THINK OF IT LIKE THIS

A path is the road from home to college. A trajectory is the road plus a timetable: when you pass each point and how fast you are going.

Joint space vs Cartesian space

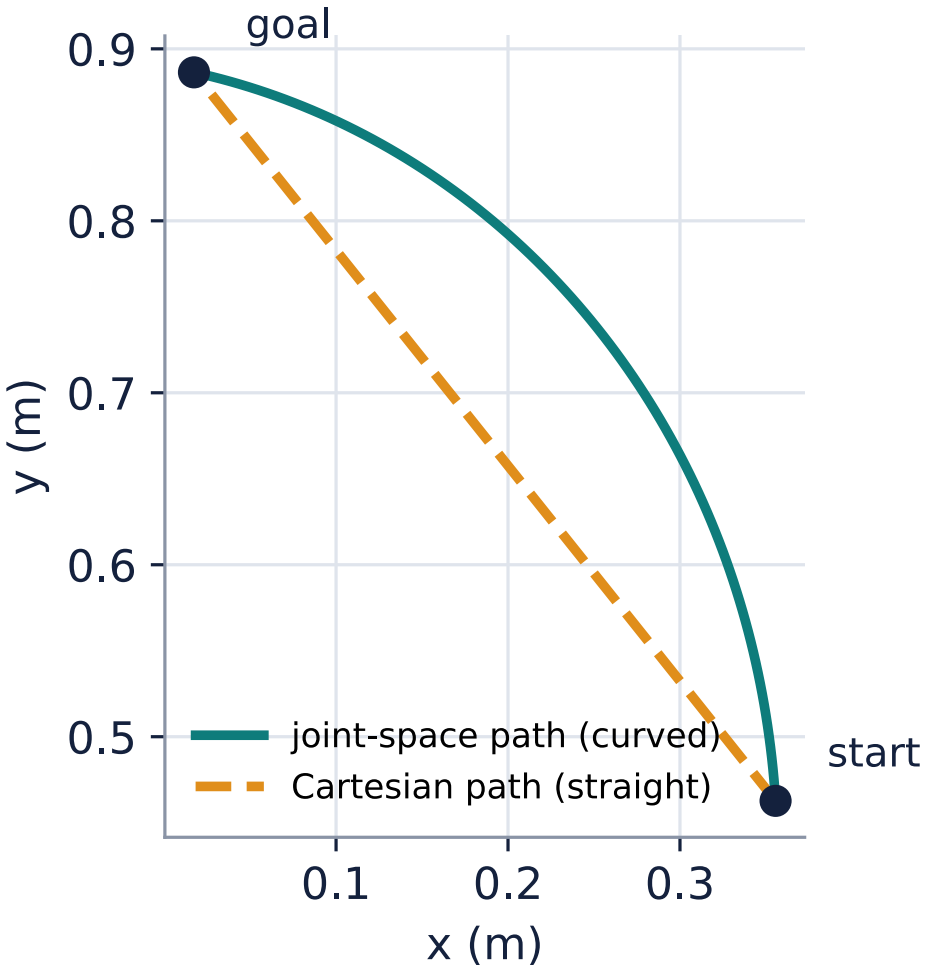
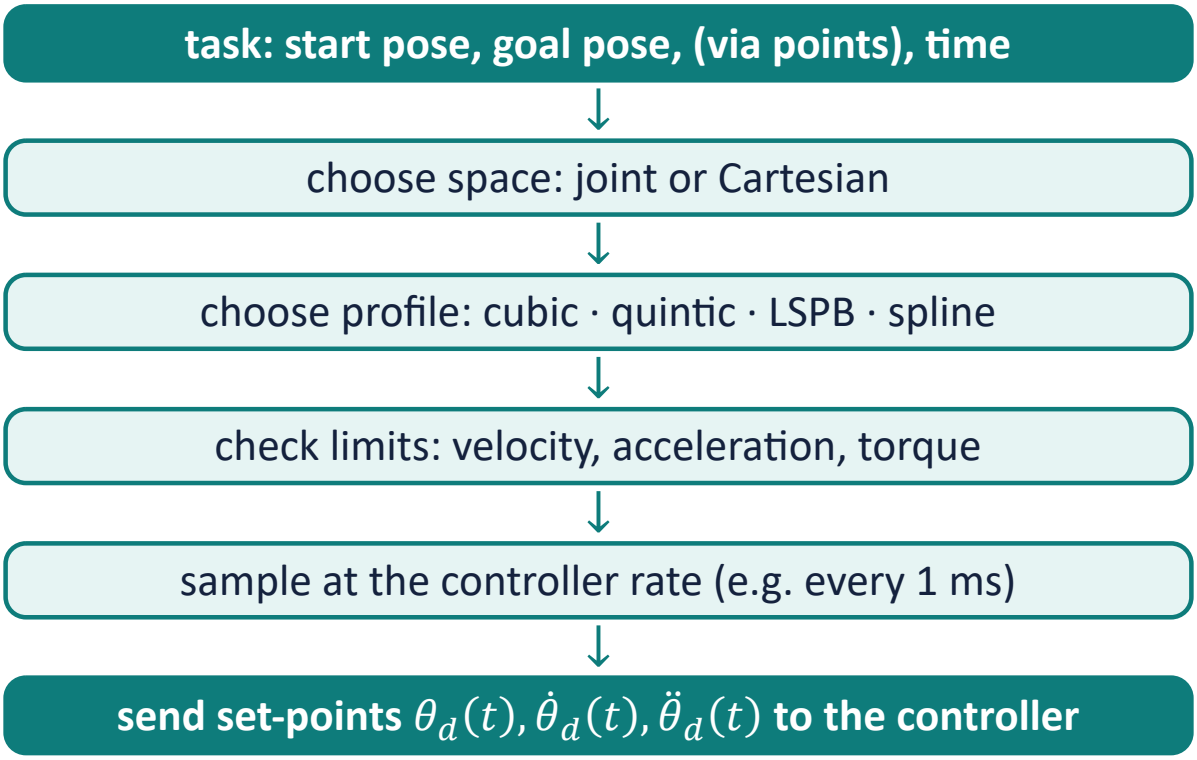


Fig. 28: Same start and goal: a joint-space plan curves; a Cartesian plan is a straight line.

	Joint-space planning	Cartesian-space planning
Plan made for	each joint angle $\theta_i(t)$	the hand position $x(t), y(t), z(t)$
Hand path	curved, not predictable	exactly as wanted (e.g. straight line)
Computation	low: no IK needed during motion	high: IK at every instant
Singularities	no IK along the way, but the motion can still pass through a singular pose: check it	path near a singularity → IK fails, joint speeds explode
Used for	pick-and-place, free moves	welding, gluing, painting along a seam



WRITE THIS IN THE EXAM

6 marks

Explain trajectory generation. Compare joint-space and Cartesian-space trajectories.

A trajectory specifies position, velocity and acceleration of each DOF as functions of time between start and goal (and via points), to be followed by the controller. **Joint space:** interpolate each $\theta_i(t)$; cheap, no repeated IK during motion, but the hand path is not controlled and the arm can still pass through singular configurations, so singularities must still be checked. **Cartesian space:** interpolate the end-effector pose; straight-line or shaped paths, needs IK at every step, may pass singularities or leave the workspace. Requirements: smoothness (continuous velocity and acceleration), and respecting velocity / acceleration / torque limits.

SYLLABUS TOPIC

5.2 Polynomial & Spline Trajectories

IN SIMPLE WORDS

Fit a smooth polynomial curve $\theta(t)$ between the start and the goal. More conditions (speed, acceleration at the ends) need a higher-degree polynomial.

Cubic polynomial (4 conditions)

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

Conditions: $\theta(0) = \theta_0, \theta(t_f) = \theta_f, \dot{\theta}(0) = 0, \dot{\theta}(t_f) = 0$ (start and stop at rest).

$$a_0 = \theta_0, \quad a_1 = 0, \quad a_2 = \frac{3(\theta_f - \theta_0)}{t_f^2}, \quad a_3 = -\frac{2(\theta_f - \theta_0)}{t_f^3}$$

Peak speed (at $t_f/2$): $\dot{\theta}_{\max} = 1.5(\theta_f - \theta_0)/t_f$ Max acceleration (at the ends): $6(\theta_f - \theta_0)/t_f^2$

Solved example 5.1 · Cubic trajectory, rest to rest

checked with sympy

Given: a joint moves from $10°$ to $70°$ in 3 s, starting and ending at rest.

STEP 1 $\theta_f - \theta_0 = 60°, t_f = 3$ s.

STEP 2 Coefficients:

$$a_0 = 10, \quad a_1 = 0$$

$$a_2 = \frac{3 \times 60}{9} = 20.000, \quad a_3 = -\frac{2 \times 60}{27} = -4.444$$

STEP 3 So $\theta(t) = 10 + 20.000\,t^2 - 4.444\,t^3$. Check at $t = 1.5$ s (half time): $\theta = 40.00°$ (exactly halfway ✓).

STEP 4 Peak velocity $= 1.5 \times 60/3 = 30.0°/s$; max acceleration $= 6 \times 60/9 = 40.00°/s^2$.

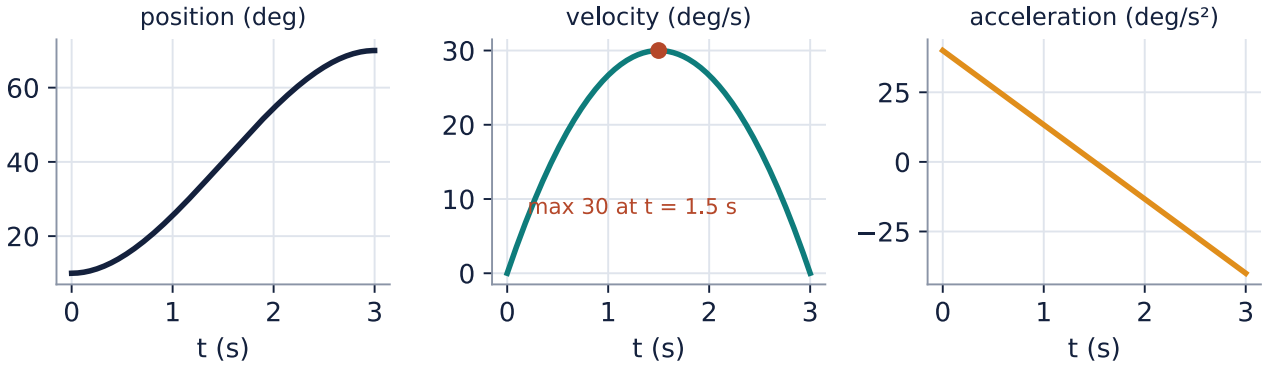


Fig. 29: Position is an S-curve; velocity is a hill; acceleration is a straight line.

ANSWER

$\theta(t) = 10 + 20t^2 - 4.444t^3$ (degrees, seconds). Peak speed 30.0 °/s at 1.5 s.

COMMON MISTAKE

A cubic's acceleration **jumps** from 0 to its maximum at the start and end, which can shake the robot. Use a quintic when smoothness matters.

Cubic with given end speeds (passing through points)

If the joint must leave at $\dot{\theta}_0$ and arrive at $\dot{\theta}_f$:

$$a_0 = \theta_0, \quad a_1 = \dot{\theta}_0$$

$$a_2 = \frac{3(\theta_f - \theta_0)}{t_f^2} - \frac{2\dot{\theta}_0}{t_f} - \frac{\dot{\theta}_f}{t_f}$$

$$a_3 = -\frac{2(\theta_f - \theta_0)}{t_f^3} + \frac{\dot{\theta}_f + \dot{\theta}_0}{t_f^2}$$

Solved example 5.2 · Cubic with non-zero end velocities

checked with sympy

Given: $\theta_0 = 0°, \theta_f = 60°, t_f = 2$ s, $\dot{\theta}_0 = 10°/\text{s}, \dot{\theta}_f = 20°/\text{s}$.

STEP 1 $a_0 = 0, a_1 = 10.$

STEP 2 $a_2 = 3(60)/4 - 2(10)/2 - 20/2 = 45 - 10 - 10 = 25.00$

STEP 3 $a_3 = -2(60)/8 + (20 + 10)/4 = -15 + 7.5 = -7.50$

ANSWER

$\theta(t) = 10t + 25t^2 - 7.5t^3$. It reaches 60° at 2 s with speed 20 °/s (checked).

Quintic polynomial (6 conditions)

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

Also fixes **acceleration** at both ends. For rest-to-rest with zero end accelerations ($D = \theta_f - \theta_0$):

$$a_0 = \theta_0, \quad a_1 = a_2 = 0, \quad a_3 = \frac{10D}{t_f^3}, \quad a_4 = -\frac{15D}{t_f^4}, \quad a_5 = \frac{6D}{t_f^5}$$

Solved example 5.3 · Quintic trajectory

checked with sympy

Given: $0° \to 60°$ in 2 s, at rest with zero acceleration at both ends.

STEP 1 $a_3 = 10(60)/8 = 75.00$, $a_4 = -15(60)/16 = -56.25$, $a_5 = 6(60)/32 = 11.250$.

STEP 2 Mid-point check ($t = 1$): $\theta = 75 - 56.25 + 11.25 = 30.00°$ (halfway ✓). Peak speed there: 56.25 °/s.

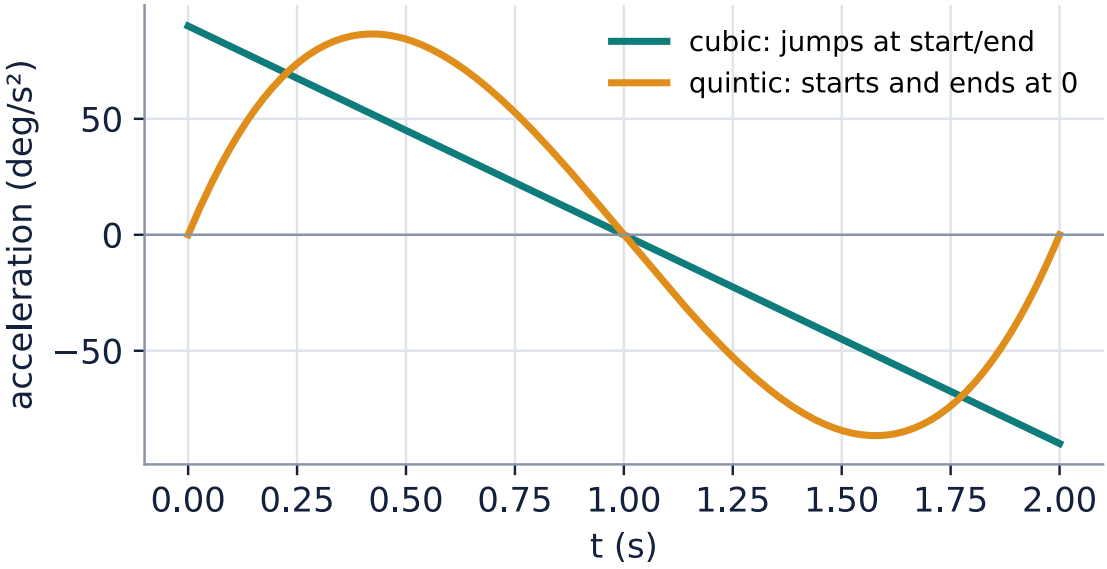


Fig. 30: Acceleration: the cubic jumps at the ends; the quintic starts and ends smoothly at zero.

ANSWER

$\theta(t) = 75t^3 - 56.25t^4 + 11.25t^5$. Smoother than a cubic, at the cost of a higher peak speed (56.25 vs 45 °/s).

Linear segment with parabolic blends (LSPB / trapezoidal)

IN SIMPLE WORDS

Speed up at constant acceleration, cruise at constant speed, slow down at constant deceleration. The velocity graph is a trapezium.

$$t_b = \frac{t_f}{2} - \frac{\sqrt{\ddot{\theta}^2 t_f^2 - 4\ddot{\theta}(\theta_f - \theta_0)}}{2\ddot{\theta}} \qquad \text{needs } \ddot{\theta} \geq \frac{4(\theta_f - \theta_0)}{t_f^2}$$

Blend (cruise) speed $V = \ddot{\theta} t_b$; angle covered in each blend $= \frac{1}{2}\ddot{\theta} t_b^2$.

Solved example 5.4 · LSPB trajectory

checked with sympy

Given: $0° \to 90°$ in 3 s, blend acceleration $\ddot{\theta} = 60°/s^2$.

STEP 1 Feasibility: minimum acceleration $= 4(90)/9 = 40°/s^2 \leq 60$ ✓

STEP 2 Blend time:

$$t_b = 1.5 - \frac{\sqrt{60^2(9) - 4(60)(90)}}{120} = 1.5 - \frac{\sqrt{10800}}{120} = 0.634 \text{ s}$$

STEP 3 Cruise speed $V = 60 \times 0.634 = 38.04°/s$; each blend covers $12.06°$.

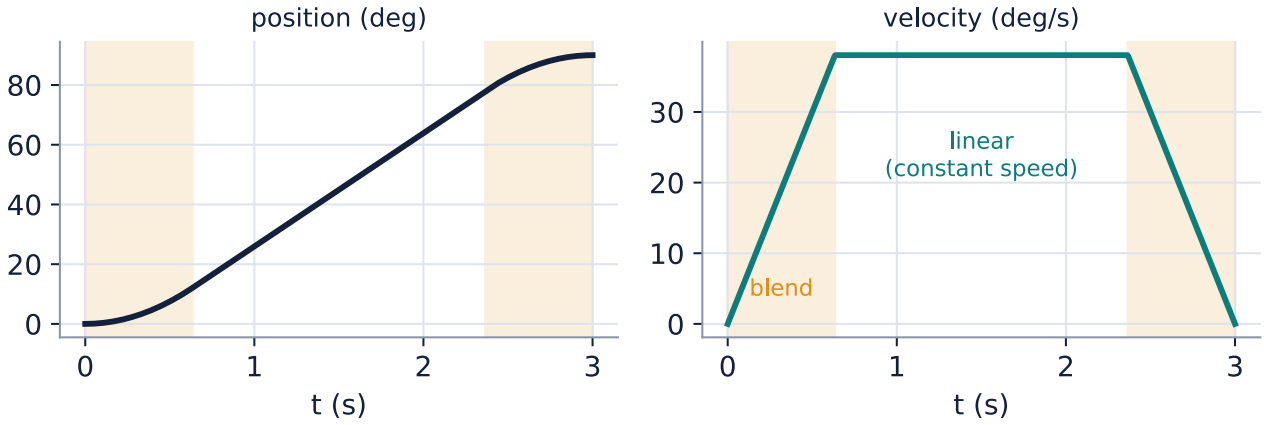


Fig. 31: Shaded = parabolic blends; middle = constant-speed straight line.

ANSWER

$t_b = 0.634$ s, cruise at 38.04 °/s from 0.634 s to 2.366 s.

Via points and splines

IN SIMPLE WORDS

To pass through intermediate points, join several cubic pieces so that position and velocity match where they meet. A smooth chain of polynomials is a spline.

Solved example 5.5 · Two cubic segments through a via point

checked with sympy

Given: $0° \to 40°$ (via) $\to 60°$, each segment 2 s, starting and ending at rest. Via velocity = average of the two slopes: $\frac{1}{2}(40/2 + 20/2) = 15.0°/s$.

STEP 1 Segment 1 ($0 \to 40$, $\dot{\theta}_0 = 0$, $\dot{\theta}_f = 15$): $\theta_1(t) = 22.5t^2 - 6.25t^3$

STEP 2 Segment 2 ($40 \to 60$, $\dot{\theta}_0 = 15$, $\dot{\theta}_f = 0$): $\theta_2(t) = 40 + 15t - 1.25t^3$

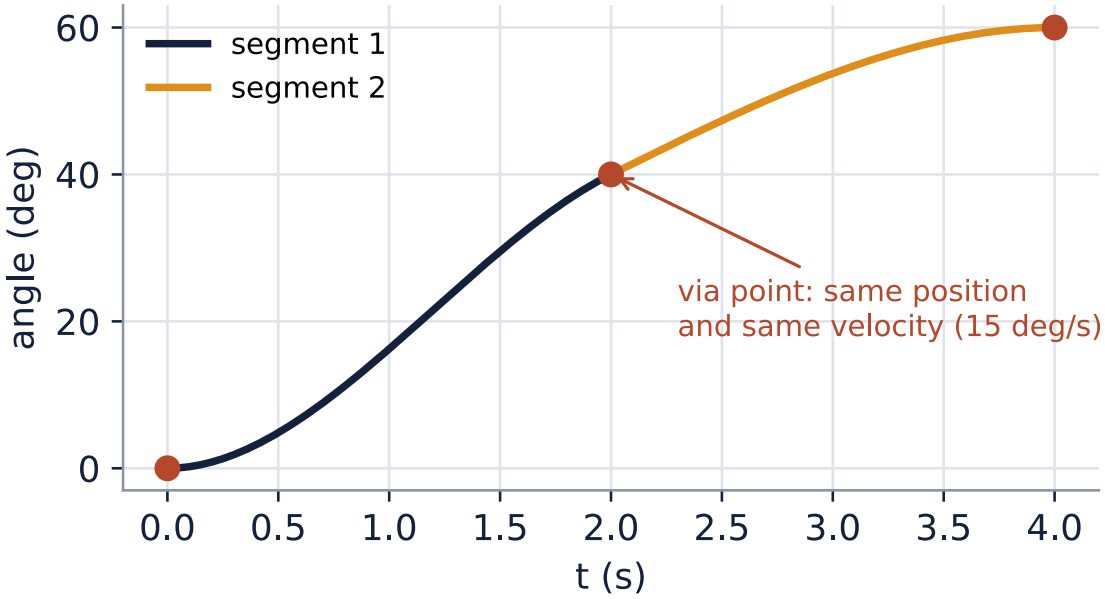


Fig. 32: The two pieces join with the same position and velocity at the via point.

ANSWER | VIA-POINT CONTINUITY

At t = 2 s, both segments give 40 degrees and 15 degrees/s. Acceleration changes from -30 to 0 degrees/s^2: position and speed are continuous (C1), but acceleration is not (not C2).

EXAM DISTINCTION

A conventional cubic interpolating spline can enforce C2 continuity at internal knots. This two-piece interpolant enforces only C1. A B-spline generally need not pass through its control points.

WRITE THIS IN THE EXAM 7 marks

Explain polynomial trajectories. Derive the cubic polynomial coefficients for rest-to-rest motion.

$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3; \dot{\theta} = a_1 + 2a_2 t + 3a_3 t^2.$ Conditions: $\theta(0) = \theta_0 \Rightarrow a_0 = \theta_0; \dot{\theta}(0) = 0 \Rightarrow a_1 = 0;$ $\theta(t_f) = \theta_f$ and $\dot{\theta}(t_f) = 0$ give $a_2 t_f^2 + a_3 t_f^3 = \theta_f - \theta_0$ and $2a_2 t_f + 3a_3 t_f^2 = 0$. Solving: $a_2 = 3(\theta_f - \theta_0)/t_f^2, a_3 = -2(\theta_f - \theta_0)/t_f^3.$

Quintic (6 conditions) also fixes end accelerations for smoother motion. **LSPB:** accelerate, cruise, decelerate (trapezoidal velocity). **Via points / splines:** join pieces with continuous position and velocity (and acceleration for cubic splines).

SYLLABUS TOPIC

5.3 Trajectory Constraints

IN SIMPLE WORDS

Every trajectory must stay inside what the robot and the job allow: limits on position, speed, acceleration, jerk and torque, plus obstacles and the time available.

Constraint	Meaning / why
Boundary conditions	start and goal positions, velocities (and accelerations)
Joint limits	$\theta_{\min} \le \theta \le \theta_{\max}$ (mechanical stops)
Velocity limit	$\lvert\dot{\theta}\rvert \le \dot{\theta}_{\max}$ (motor speed)
Acceleration limit	$\lvert\ddot{\theta}\rvert \le \ddot{\theta}_{\max}$ (motor torque, gear strength)
Jerk limit	smooth changes of acceleration → less vibration and wear
Torque limit	$\lvert\tau\rvert \le \tau_{\max}$ using the dynamics of Unit IV
Continuity	no jumps in position or velocity (and acceleration if needed) at via points
Obstacles / workspace	path must avoid collisions and stay reachable
Time	finish within the cycle time; ideally as fast as the limits allow

Solved example 5.6 · Is the cubic of Example 5.1 allowed?

checked with sympy

Limits: speed ≤ 40 °/s, acceleration ≤ 50 °/s².

STEP 1 Peak speed 30.0 °/s ≤ 40 ✓; peak acceleration 40.00 °/s² ≤ 50 ✓.

STEP 2 If the move had to finish in 2 s: speed $= 1.5(60)/2 = 45$ °/s ✗ (too fast). Minimum time from speed: $t_f \geq 1.5(60)/40 = 2.25$ s; from acceleration: $t_f \geq \sqrt{6(60)/50} = 2.68$ s.

ANSWER | The 3-second cubic satisfies the stated limits. Within rest-to-rest cubics, the minimum duration under ONLY speed and acceleration limits is 2.68 seconds. Also check torque, jerk and hardware limits.

WRITE THIS IN THE EXAM

6 marks

Explain the constraints considered in trajectory planning.

Boundary conditions (start/goal position, velocity, acceleration); joint position limits; velocity and acceleration limits of actuators; jerk limits for smoothness; torque limits from the dynamic model $\tau = M\ddot{q} + C\dot{q} + G$; continuity at via points; obstacle avoidance and workspace limits; cycle-time requirements. A trajectory satisfying all of them is **feasible**; the fastest feasible one is **time-optimal**.

SYLLABUS TOPIC

5.4

Control Architectures

IN SIMPLE WORDS

The controller compares where the robot should be with where it actually is (measured by sensors) and sets the motor torques to remove the difference.

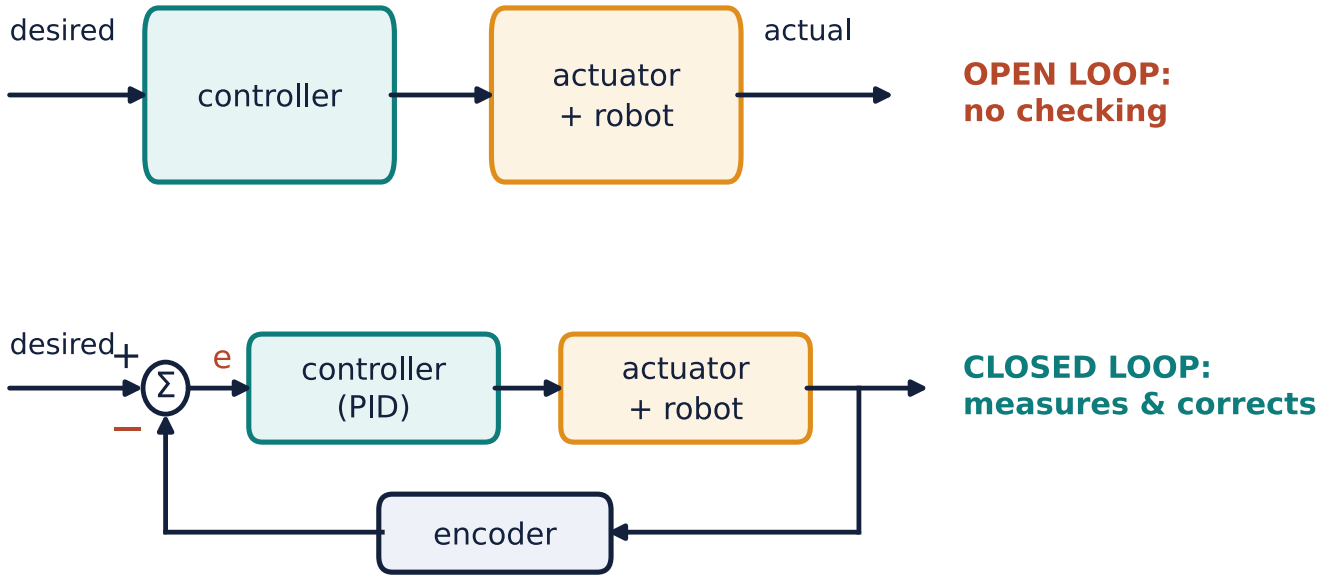


Fig. 33: Open loop does not check the result; closed loop measures and corrects.

The PID controller (most common, one per joint)

$$\tau = K_p\, e + K_i \int e\, dt + K_d\, \dot{e}, \qquad e = \theta_d - \theta$$

Term	Does	Too much of it
P (proportional)	pushes harder the bigger the error	overshoot, oscillation
I (integral)	removes the remaining steady error (e.g. gravity sag)	slow wobbles, wind-up
D (derivative)	brakes on the error's rate: damping	noise amplification

Control architectures used in robots

Architecture	Idea
Independent joint control	each joint has its own PID; other joints treated as disturbances. Simple, most common in industry
PD + gravity compensation	$\tau = K_p e + K_d \dot{e} + G(q)$: removes the gravity sag
Computed torque / inverse dynamics	uses the full model to cancel the dynamics (topic 5.5)
Hierarchical	planner → trajectory generator → joint servo loops (fast inner loop, slower outer loop)
Force / impedance control	controls contact force or stiffness: grinding, assembly, cobots
Adaptive / robust	adjusts to unknown payloads or model errors

Solved example 5.7 · PD control of a one-link arm, and the gravity problem checked with sympy

Arm: $m = 2$ kg, $l = 0.5$ m ($I = 0.50$ kg·m², $mgl = 9.810$ N·m). Target $30°$, start at $0°$.

STEP 1 **Plain PD** $\tau = K_p(\theta_d - \theta) - K_d\dot{\theta}$ with angles in **radians**: $K_p = 50$ N·m/rad, $K_d = 5$ N·m·s/rad. At rest the controller torque must balance gravity: $K_p(\theta_d - \theta) = mgl\cos\theta$, with $\theta_d = 30° = 0.5236$ rad.

STEP 2 Solving this, the arm settles at $\theta = 19.40°$, so a **steady-state error** of 10.60° remains.

STEP 3 **Fix:** add an I term, add gravity compensation $+G(q)$, or use computed torque (next topic).

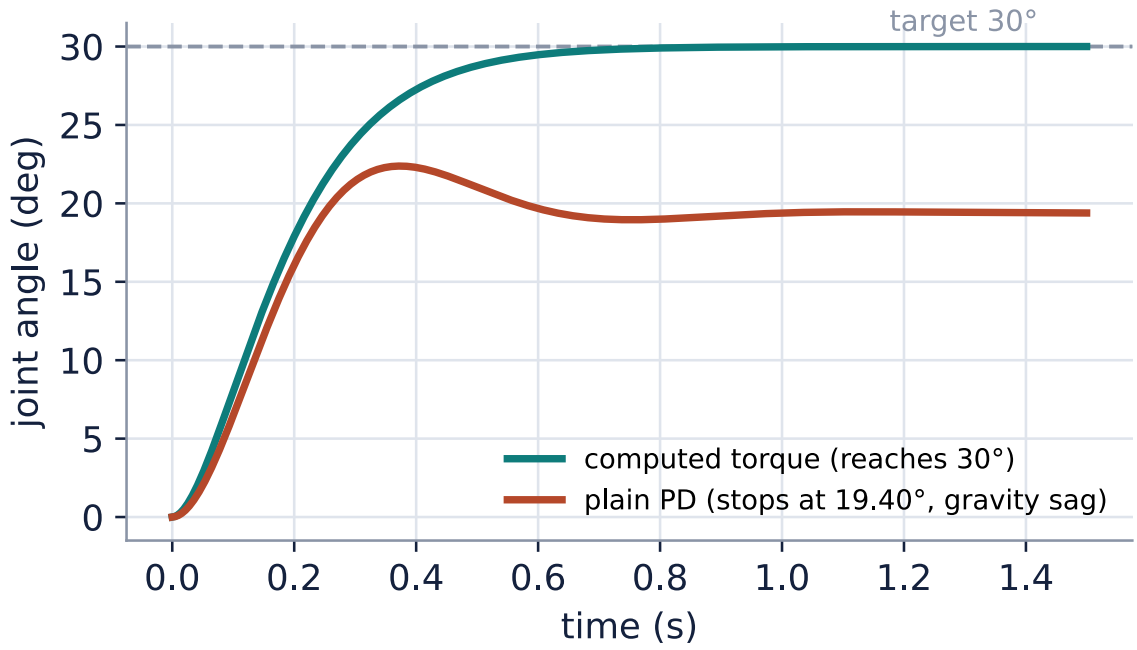


Fig. 34: Simulated response: plain PD sags under gravity; computed torque reaches the target.

ANSWER

Plain PD stops at 19.40° (error 10.60°). Gravity must be compensated for accurate positioning.

WRITE THIS IN THE EXAM

6 marks

Explain control architectures for robot manipulators. Explain PID control.

Open loop: commands without measuring (simple, inaccurate). **Closed loop:** sensors (encoders) feed back the actual position; error $e = \theta_d - \theta$ drives the controller. **PID:** $\tau = K_p e + K_i \int e + K_d \dot{e}$; P reacts to error, I removes steady-state error, D adds damping.

Architectures: independent joint (decentralised) PID; PD + gravity compensation; computed-torque (model-based, centralised); hierarchical (task planner → trajectory → servo); force / impedance control; adaptive / robust control.

SYLLABUS TOPIC

5.5

Feedback Linearization

IN SIMPLE WORDS

Computed torque uses a robot dynamics model to compensate nonlinear effects. With an accurate model and ideal actuation, the tracking-error equation becomes linear. Model errors, friction, delays and torque limits can prevent exact cancellation.

THINK OF IT LIKE THIS

Noise-cancelling headphones estimate unwanted sound and counteract it. Similarly, computed torque predicts forces from inertia, coupling and gravity. Feedback handles residual errors; cancellation is approximate when the model or actuators are imperfect.

The robot obeys $\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q)$. Choose the torque:

$$\tau = M(q)\,u + C(q,\dot{q})\,\dot{q} + G(q), \qquad u = \ddot{q}_d + K_d\,\dot{e} + K_p\,e$$

Substituting gives $M\ddot{q} = Mu$, so $\ddot{q} = u$, and the error obeys

$$\ddot{e} + K_d\,\dot{e} + K_p\,e = 0$$

a simple, linear, decoupled equation for every joint. Choose $K_p = \omega_n^2$, $K_d = 2\zeta\omega_n$.

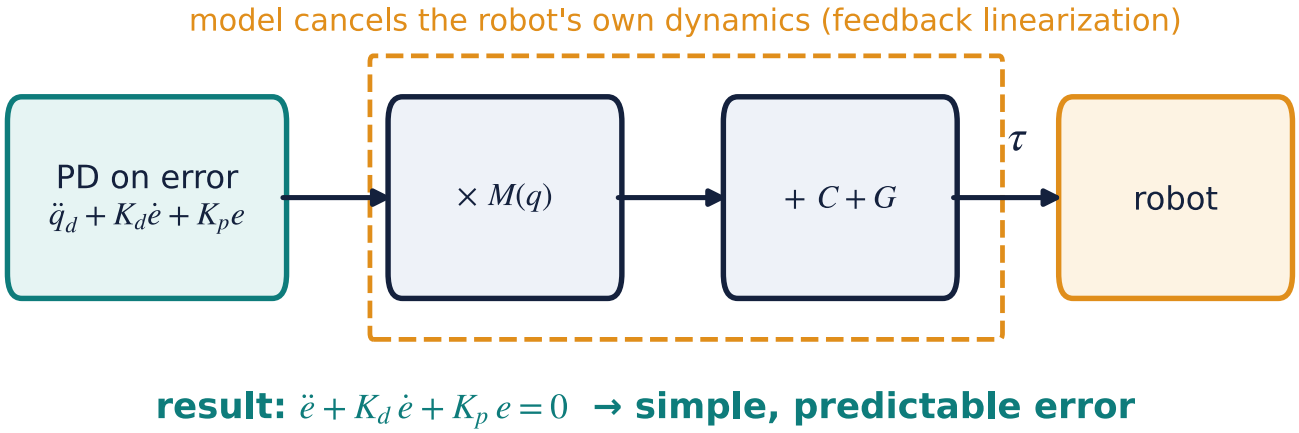


Fig. 35: Computed-torque (feedback linearization) control.

Solved example 5.8 · Choosing the gains checked with sympy

Want: error to settle within about 0.6 s with **no overshoot** → natural frequency $\omega_n = 10$ rad/s, damping $\zeta = 1$ (critical). For critical damping the 2 % settling time is $\approx 5.8/\omega_n = 0.58$ s (simulation: 0.58 s).

STEP 1 $K_p = \omega_n^2 = 100\ \text{s}^{-2}$

STEP 2 $K_d = 2\zeta\omega_n = 20\ \text{s}^{-1}$ (these gains act on the **acceleration** u, so their units differ from plain-PD gains, which multiply by the inertia to give N·m)

STEP 3 For the one-link arm: $\tau = I\left(\ddot{\theta}_d + 20\dot{e} + 100e\right) + mgl\cos\theta$.

ANSWER

$K_p = 100, K_d = 20$. In the simulation above (teal curve), the arm reaches 30° with no sag and no overshoot.

REMEMBER | IDEAL MODEL VS REAL ROBOT

Model cancellation yields a linear tracking-error equation only for sufficiently accurate dynamics and unsaturated actuation. Payload errors, friction, sensor delays and limits can worsen real tracking.

WRITE THIS IN THE EXAM **7 marks**

Explain feedback linearization (computed-torque control) of a manipulator.

Robot model: $\tau = M(q)\ddot{q} + C\dot{q} + G$. Control law: $\tau = M(q)u + C\dot{q} + G$ with $u = \ddot{q}_d + K_d\dot{e} + K_p e$. Substituting gives $\ddot{q} = u$, so the closed loop is $\ddot{e} + K_d\dot{e} + K_p e = 0$: linear, decoupled, with poles set by $K_p = \omega_n^2, K_d = 2\zeta\omega_n$. Draw the block diagram

Merit: linear, decoupled error dynamics under ideal assumptions. Limits: uncertain loads, friction, delay and motor saturation. Validate tracking on the real system.

5.6

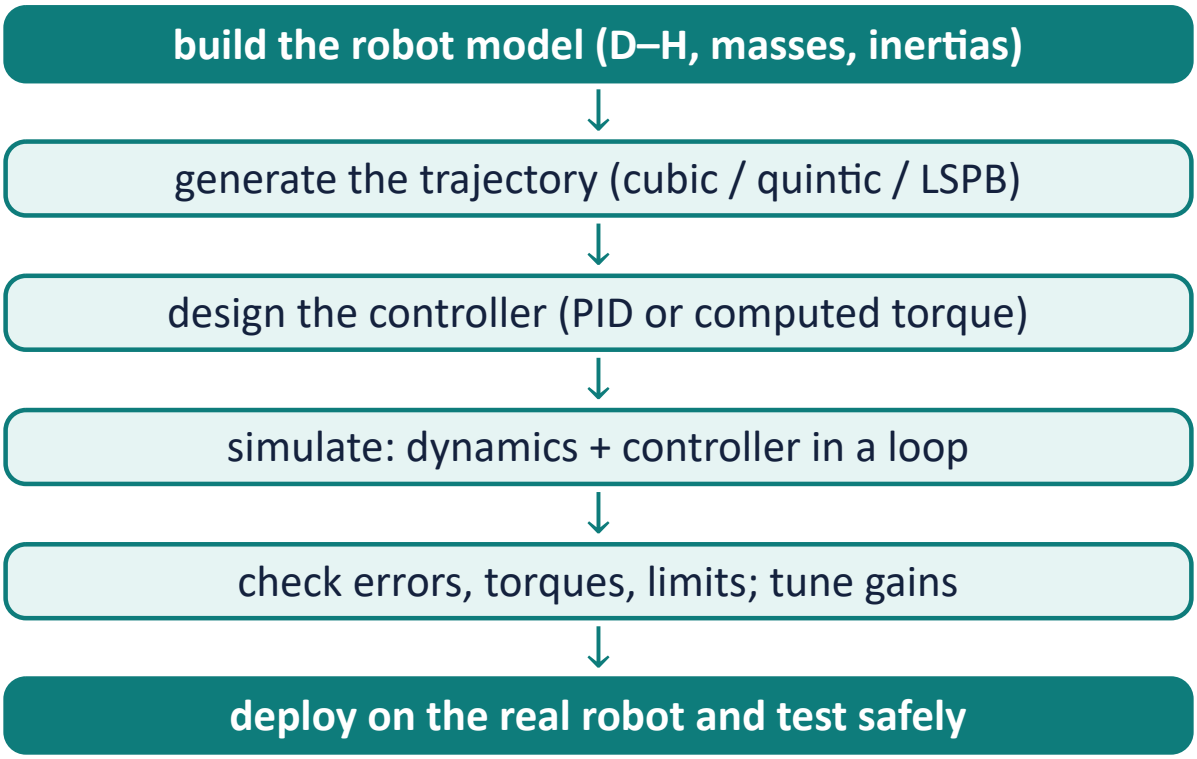
SYLLABUS TOPIC

Simulation & Practical Implementation

IN SIMPLE WORDS

Before a real robot moves, the whole chain (trajectory → controller → robot model) is simulated to check it. Then it runs on real hardware with real sensors, limits and timing.

Simulation workflow



Software: MATLAB / Simulink (Robotics System Toolbox), Python (NumPy, roboticstoolbox), ROS + Gazebo / MoveIt, CoppeliaSim, Webots.

Practical implementation issues

Issue	What to do
Sampling rate	joint servo loops run fast (about 1 kHz); planners slower
Sensor noise	filter velocity estimates; keep K_d moderate
Actuator saturation	respect torque limits; anti-wind-up for the I term
Friction, backlash, gears	friction compensation, stiff gearboxes
Model errors / payload changes	robust or adaptive control, payload identification
Safety	speed limits, emergency stop, collision detection, safety zones
Calibration	measure the real D–H parameters for accuracy

WRITE THIS IN THE EXAM

6 marks

Explain the simulation and practical implementation of a robot trajectory controller.

Simulation: model the robot (kinematics + dynamics), generate the reference trajectory, implement the controller, integrate $\ddot{q} = M^{-1}(\tau - C\dot{q} - G)$ numerically, and study tracking error, torques and limits; tune gains (e.g. in MATLAB/Simulink, ROS-Gazebo).

Practical issues: digital implementation at a fixed sampling rate; encoder resolution and noise; actuator saturation and integral

wind-up; friction and backlash; model uncertainty and payload variation; safety (e-stop, limits). Hardware: controller, motor drives, encoders, PLC / industrial PC.

Unit V · Check yourself

1. Path vs trajectory: what is the difference?
2. How many conditions does a cubic satisfy? A quintic?
3. Cubic rest-to-rest: when is the speed highest?
4. What is the velocity graph of LSPB shaped like?
5. What must match at a via point?
6. Which PID term removes steady-state error?
7. After feedback linearization, what equation does the error obey?

ANSWERS

1 · A trajectory adds timing (speed, acceleration). 2 · 4; 6. 3 · At half the time. 4 · A trapezium. 5 · Position and velocity (and acceleration for splines). 6 · I (integral). 7 · $\ddot{e} + K_d\dot{e} + K_p e = 0$.

Unit V · Question bank

No.	Question	M
1	Explain trajectory generation; compare joint and Cartesian space.	6
2	Derive the cubic polynomial coefficients for rest-to-rest motion.	7
3	Numerical: cubic coefficients, peak velocity and acceleration.	7
4	Explain the quintic trajectory; why is it smoother than a cubic?	6
5	Explain LSPB; find the blend time for given data.	7
6	Explain via points and spline trajectories.	6
7	Explain the constraints in trajectory planning.	6
8	Explain open- and closed-loop control and PID for a joint.	6
9	Explain control architectures used for manipulators.	6
10	Explain feedback linearization / computed-torque control with a block diagram.	7
11	Explain simulation and practical implementation issues.	6