PPP MANIPULATOR

Final Project Report

Introduction to Robotics

Mechanical Engineering

National University of Science and Technology

Members:

- Nicolas Miranda
- Juan Bogado
- Juan Chavez
- Saul Ferreira

Supervisors:

- Liang Shuhao (Jason)
- Marnel Patrick Junior Altius

Part 1: Introduction

Application

The difficulty of the project largely depends on its scope and intended application. Proper utilization of references and prior examples can significantly reduce the required effort, as they allow leveraging existing solutions and tested approaches. This project focuses on designing a practical, functional system that is simple enough for prototyping, yet demonstrates the core concepts effectively.

The system targets automated recycling classification using a PPP robot with a two-finger parallel gripper. It classifies small objects into three categories—metal, plastic, and paper—using cost-effective, non-vision sensors. Metal items are identified by an inductive proximity sensor, while plastic and paper are differentiated using a combination of capacitive dielectric sensing and mass measurement via a 5 kg load cell. All sensors are integrated into a fixed sensing station, which ensures stable distance, controlled grounding, low noise conditions, and repeatable force application, improving detection reliability during classification.

Robot Title and Type

The robot, titled *Automated Recycling Classifier (ARC)*, is a PPP Cartesian manipulator equipped with a two-finger parallel gripper. It employs three prismatic joints for translational motion along the X, Y, and Z axes. The use of a PPP configuration improves mechanical simplicity, supports straightforward simulation and kinematic modeling, and enables rapid prototype construction.

Problem and Task Description

Manual sorting in recycling facilities is inefficient, inconsistent, and labor-dependent. This project automates the separation of materials by enabling the robot to pick items from the sensing zone, classify them, and deposit them into the correct bin. All detection takes place at the bench-mounted fixed sensing station, where the object's metallic response, dielectric properties, and mass are measured before sorting.

The inclusion of a load cell provides an additional physical parameter that improves robustness when distinguishing between paper and plastic, which often exhibit overlapping dielectric characteristics.

Functional and Performance Requirements

The ARC robot must autonomously complete pick–classify–place cycles. Metal detection is executed using the inductive sensor. Plastic and paper differentiation is achieved through sensor fusion, combining capacitive dielectric measurements with mass data obtained from a 5 kg load cell mounted beneath the sensing platform. The classification process occurs at the fixed sensing station, ensuring repeatable positioning and consistent load application for reliable signal capture.

The gripper must handle objects of varying geometry without tool changes, and the system must consistently allocate objects into three output bins.

Performance Goals

- $\geq 90\%$ classification accuracy
- ≤ 5 seconds per item cycle time
- End-effector repeatability within ± 2 mm
- Support for object masses up to 5 kg, matching the load cell measurement range

Operating Environment and Constraints

The platform is designed for benchtop operation without machine vision or external lighting. Objects must fit within the robot's reachable workspace and be small enough for the parallel gripper. The fixed sensing station requires stable grounding, controlled contact force, and rigid mounting of the load cell to ensure accurate mass readings. Calibration of the capacitive sensor and load cell must be conducted prior to operation. Environmental factors such as humidity may influence dielectric measurements.

Constraints

- Inductive sensing limited exclusively to metal detection
- Capacitive readings sensitive to geometry, moisture, and surface area
- Load cell accuracy influenced by vibration and dynamic loading
- Limited detection capability for composite or multilayer materials
- Workspace constrained by PPP travel limits

Motivation

The objective of the project is to demonstrate an affordable and compact robotic sorting system that improves sorting consistency while reducing manual labor. The integration of mass sensing enhances classification reliability without introducing complex vision systems. Automation in recycling environments reduces injuries, increases throughput stability, and enables scalable material handling at low operational cost. Institutions such as research labs, educational programs, and small recycling stations benefit directly from such a system.

Cost, Design, and Technical Constraints

Because vision is excluded from the design, material identification relies on inductive sensing, capacitive dielectric measurement, and low-cost force-based mass estimation using a 5 kg load cell. The gripper must interface with all materials without custom attachments, and motion control must remain within safe velocity and acceleration boundaries. All decisions are made through real-time sensor readings processed by a controller.

Design considerations prioritize low-cost hardware for industrial application: inductive and capacitive sensors plus a load cell (combined cost $<$ USD 40), a PLC, and aluminum and stainless-steel structural materials. Technical limitations include sensor noise, dielectric overlap between materials, load cell drift, and inductive sensing limited to conductive metals.

Physical Assumptions Used in Modeling

The manipulator is modeled assuming rigid-body links with negligible flex or structural deformation, enabling simplified dynamic evaluation. Prismatic joints are assumed to follow linear velocity trajectories without significant backlash or friction losses. The parallel gripper is modeled as capable of secure handling of objects up to 5 kg, without slippage under normal acceleration profiles.

The load cell is modeled as a linear elastic element with negligible hysteresis after calibration. Sensor outputs are considered deterministic once calibrated at the fixed station, allowing threshold-based classification and simple sensor fusion logic without probabilistic modeling at this stage.

Additional Considerations

Future enhancements to the ARC platform may include machine vision for improved material recognition, machine learning-based adaptive classification combining dielectric and mass features, or increasing the number of detectable material categories. The integration of conveyor-based feed systems may also enable continuous, high-throughput sorting rather than batch-based processing. Optional end-effector-mounted force sensors could be explored to estimate mass during grasping, although this introduces additional complexity in control and dynamic compensation.

Overall Robot Diagram

The ARC system consists of a PPP gantry structure with three linear axes, a parallel gripper, and a centralized fixed sensing station containing the inductive sensor, capacitive sensor, and a load-cell-based weighing platform. The microcontroller manages sensor fusion and classification logic, while the motion controller executes programmed trajectories. The processed items are sorted into three bins according to material classification.

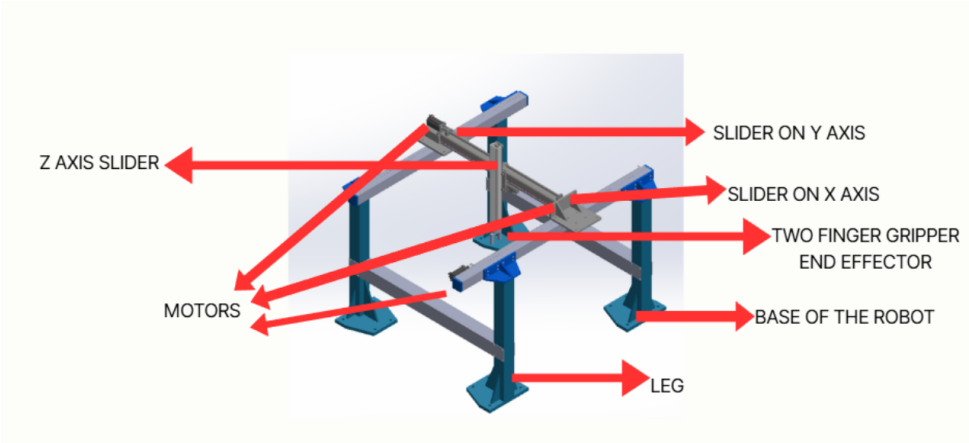

Figure 1: CAD Drawings

Figure 2: Labeled robot diagram

Manipulator Workspace

The operational workspace is defined by the gantry rails, forming a rectangular volume approximately 1000 mm × 650 mm × 410 mm. The fixed sensing station, weighing platform, and output bins are arranged within this envelope to reduce motion travel and maintain short cycle times. Objects are transported from the pick region to the sensing location, weighed and classified, and then routed to the appropriate disposal zone.

Design Specifications

CAD Drawings

All dimensions are in millimeters. The drawing scale is 4.9 relative to the robot used in this project.

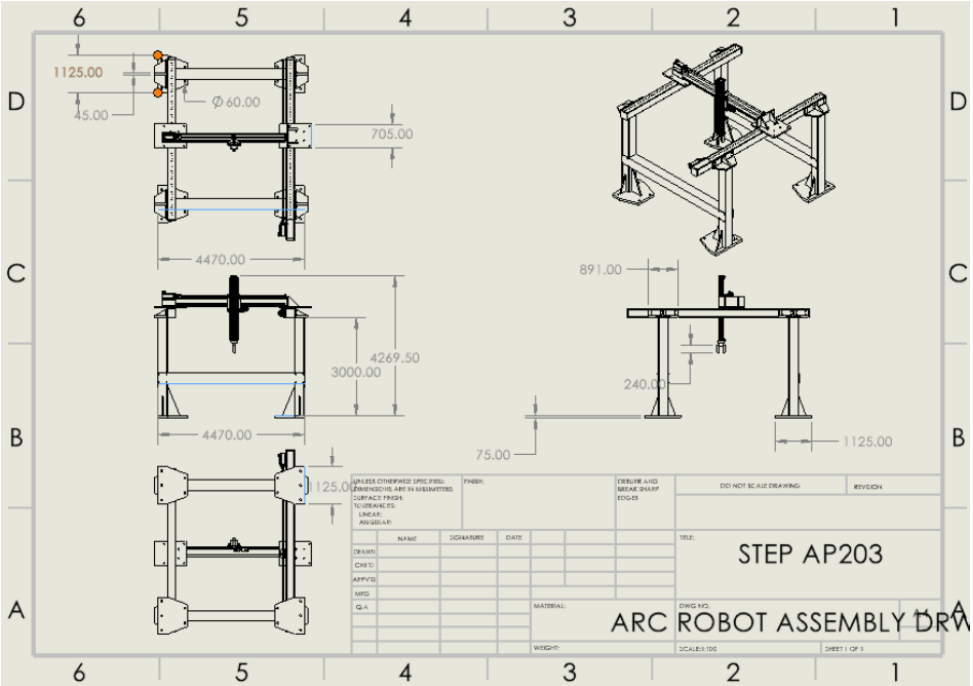

Figure 3: CAD Drawings

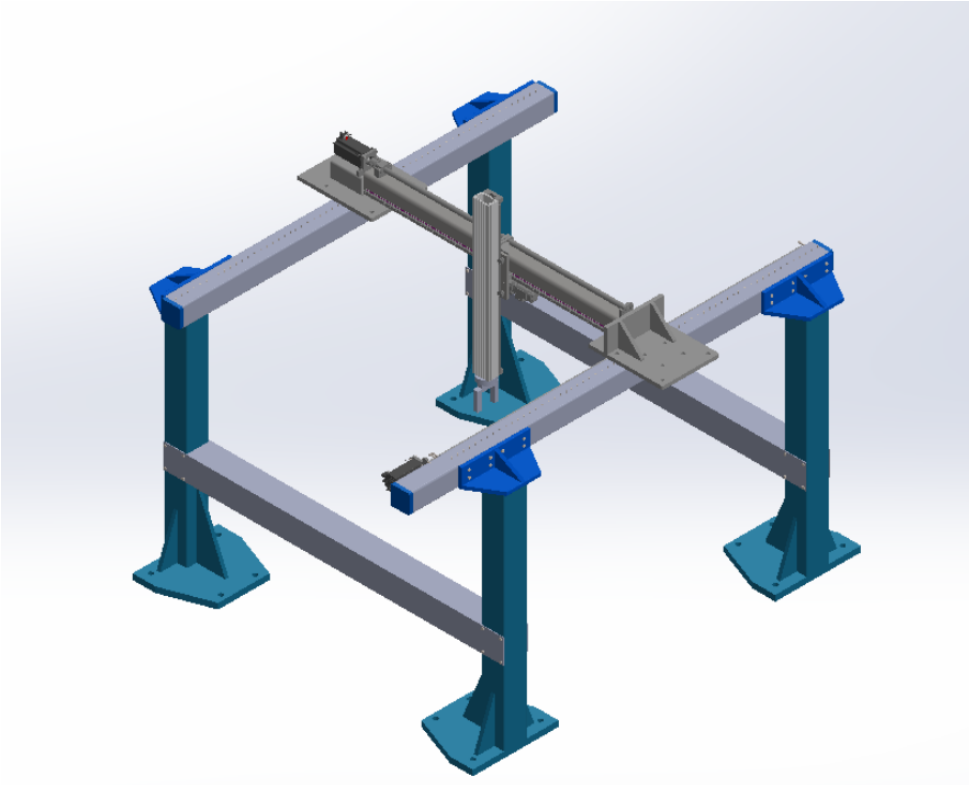

Figure 4: Cad File

Part 2: Kinematic Structure

Robot Joints and Kinematic Structure

The robot is designed as a PPP manipulator, meaning it has three prismatic joints arranged orthogonally. This configuration enables simple translational motion in the three independent directions (X, Y, Z), ensuring that the end-effector can reach any point within a rectangular workspace.

Reasoning for Design

Three prismatic joints were selected because they provide the minimum degrees of freedom required for full translational positioning in 3D space. The absence of rotational joints simplifies kinematic equations, avoids singularities, and reduces control complexity. This structure is ideal for pick-and-place tasks, material handling, and environments where end-effector orientation is fixed or non-critical.

Joint Types, Axes, and Limits

- **Joint 1 (q_1):** Prismatic along the Y axis (vertical). Critical due to gravitational load. Typical limits: $q_1 \in [q_{1,\min}, q_{1,\max}]$.
- **Joint 2 (q_2):** Prismatic along the X axis (horizontal). Provides lateral positioning. Limits: $q_2 \in [q_{2,\min}, q_{2,\max}]$.
- **Joint 3 (q_3):** Prismatic along the Z axis (depth). Provides forward/backward positioning. Limits: $q_3 \in [q_{3,\min}, q_{3,\max}]$.

Forward Kinematic Model

Frame assignments:

- Base frame $\{0\}$ fixed at the origin.
- Frame $\{1\}$ translates along Y.
- Frame $\{2\}$ translates along X.
- Frame $\{3\}$ translates along Z.
- End-effector frame attached to the gripper.

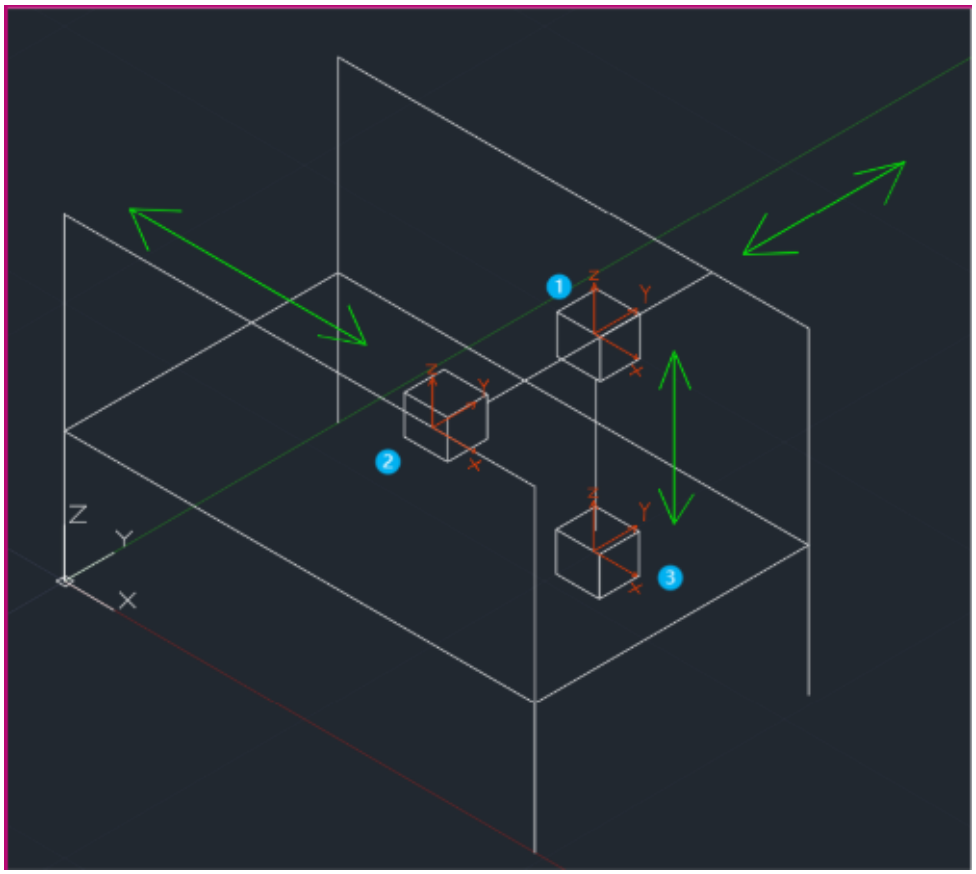

Figure 5: Schematic Diagram

The end-effector position is simply:

$$x = q_2, \quad y = q_1, \quad z = q_3.$$

The homogeneous transformation from base to end-effector is:

$$T_{0\,EE} = \begin{bmatrix} 1 & 0 & 0 & q_2 \\ 0 & 1 & 0 & q_1 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Modified DH Parameters

Since this is a PPP robot:

- All twist angles $\alpha_i = 0$.
- All link lengths $a_i = 0$.
- θ_i values are constant.
- d_i corresponds to q_1, q_2, q_3.

Inverse Kinematic Model

The IK is direct:

$$q_1 = y_d, \quad q_2 = x_d, \quad q_3 = z_d.$$

Joint limits must be satisfied:

$$q_i \in [q_{i,\min}, q_{i,\max}].$$

Velocity and acceleration limits:

$$|\dot{q}_i| \leq \dot{q}_{i,\max}, \qquad |\ddot{q}_i| \leq \ddot{q}_{i,\max}.$$

Gravity compensation on q_1 is recommended:

$$F_{g,1} = m_{\text{eff},1} g.$$

Extended Inverse Kinematic (IK) Model

For the PPP manipulator (three mutually orthogonal prismatic joints), the end-effector orientation is fixed and motion is purely translational. The IK solution is direct and exact.

Analytical Solution

Given a desired Cartesian target position (x_d, y_d, z_d) expressed in the base frame:

$$q_1 = y_d, \qquad q_2 = x_d, \qquad q_3 = z_d.$$

This yields an analytical, closed-form, globally valid solution with no singularities.

Joint Priorities and Physical Considerations

- q_1 **(vertical axis)**: Most critical due to gravitational load and safety. Requires smoother trajectories and torque/force considerations.
- q_2 **and** q_3 **(horizontal axes)**: Dynamically less restrictive; can sustain higher allowable velocities and accelerations.
- Joint priorities may be implemented through weighted objective functions in task-space or joint-space control.

IK Constraints

Joint limits

$$q_1 \in [q_{1,\min}, q_{1,\max}], \qquad q_2 \in [q_{2,\min}, q_{2,\max}], \qquad q_3 \in [q_{3,\min}, q_{3,\max}].$$

The target is feasible only if (x_d, y_d, z_d) lies inside the rectangular parallelepiped defined by these limits.

Velocity and acceleration limits

$$|\dot{q}_i| \leq \dot{q}_{i,\max}, \qquad |\ddot{q}_i| \leq \ddot{q}_{i,\max}.$$

These limits constrain the achievable trajectories during motion execution.

Collision and safety constraints Avoid operating close to hard limits or obstacles; incorporate safe margins when generating trajectories.

Stability and Robustness Considerations

- **Filtering and anti-jitter**: Apply smoothing (e.g., low-pass filters or spline interpolation) to avoid abrupt changes caused by noise.
- **Soft saturations**: Use continuous saturation functions to prevent discontinuities when enforcing limits.
- **Task-space control stability**: Since J is constant and full rank, a PID or PD controller with feedforward ensures practical stability.
- **Gravity compensation**: For the vertical joint,

  $$F_{g,1} = m_{\text{eff},1} \cdot g,$$

  improves accuracy and reduces steady-state error.

Differential Kinematic Model

The differential model relates joint velocities to Cartesian velocities of the end-effector.

Jacobian:

$$J = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad x = [x, y, z]^T, \quad q = [q_1, q_2, q_3]^T.$$

Velocities:

$$\dot{x} = J\,\dot{q}, \qquad \dot{q} = J^{-1}\dot{x}.$$

Since J is constant and full rank, inversion is stable.

Accelerations:

$$\ddot{x} = J\,\ddot{q}.$$

No $\dot{J}$ terms appear because J is constant.

Differential Control with Priorities

Weighted task-space mapping may be used with $W_x = \text{diag}(w_x, w_y, w_z)$.

Limits and Stability

Saturate $\dot{q}$ and $\ddot{q}$ to respect actuator limits. Constant full-rank J ensures numerical stability. Jacobian:

$$J = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Forward differential kinematics:

$$\dot{x} = J\,\dot{q}.$$

Inverse differential kinematics:

$$\dot{q} = J^{-1}\dot{x}.$$

Since J is constant and full rank, the inversion is stable.

Part 3: Workspace Analysis

Reachable vs Dexterous Workspace

For the PPP robot, the reachable workspace is a rectangular parallelepiped defined by the maximum ranges of all three prismatic joints. Since the robot has no rotational joints, the end-effector orientation is fixed. Therefore, the dexterous workspace coincides with the reachable workspace, but without orientation freedom.

Analytically:

$$x \in [0, Q_{2\,\max}], \quad y \in [0, Q_{1\,\max}], \quad z \in [0, Q_{3\,\max}].$$

This produces a simple box-shaped workspace.

Numerical Workspace Plot

A numerical simulation confirms that the analytical workspace matches the computed workspace.

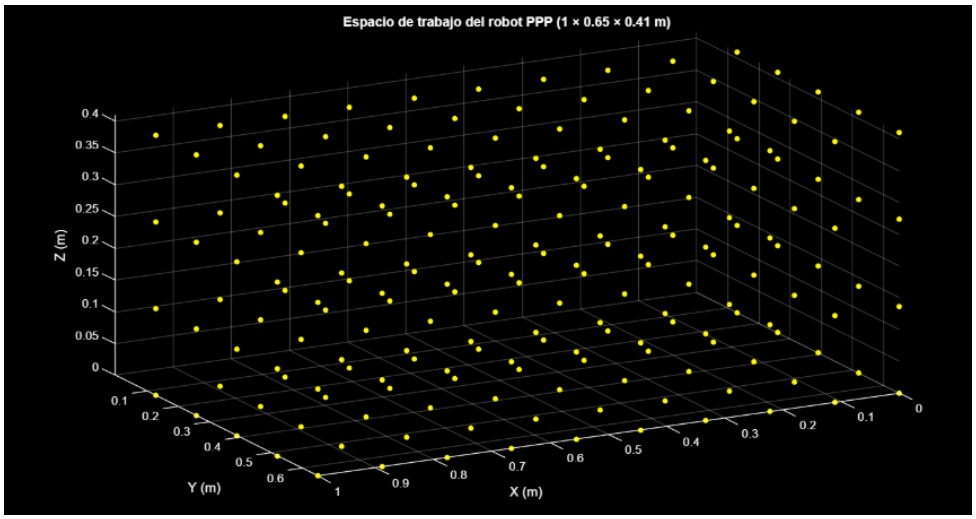

Figure 6: Workspace Plot

Jacobian-Based Singularity Identification

The Jacobian is:

$$J = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Since it is constant and full rank , no singularities occur.

Classification of Singularities

A PPP robot has no internal singularities. Only mechanical limits at the boundaries restrict motion.

Workspace Limitations and Implications

- Rectangular workspace limited by joint ranges.
- No orientation freedom.
- Not suitable for orientation-dependent tasks.
- Well-suited for structured pick-and-place applications.
- Handling capacity depends on gripper geometry.
- Limited versatility due to absence of rotational DOF.

Part 4: Dynamic Analysis and Joint Torque Requirements

Modeling Method

The dynamic modeling of the PPP Cartesian robot is carried out using a Newton–Euler formulation applied independently to each prismatic axis. Due to the orthogonal and decoupled structure of the Cartesian robot, coupling terms between axes are negligible, allowing each joint to be modeled as a translational mass subjected to inertial, gravitational, frictional, and external forces.

This approach is widely adopted in industrial gantry and Cartesian robots due to its computational simplicity and sufficient accuracy for actuator sizing.

Full Dynamic Equation

For each prismatic axis, the translational dynamic equation is expressed as:

$$F_{\text{tot}} = m \cdot a + F_{\text{grav}} + F_{\text{fric}} + F_{\text{ext}}$$

where:

- m is the equivalent moving mass including payload,
- a is the linear acceleration,
- $F_{\text{grav}} = m \cdot g$ (only for the Z-axis),
- $F_{\text{fric}} = \mu \cdot m \cdot g$,
- F_{ext} represents external process forces.

The required actuator torque is obtained from the total force using the drive kinematics:

$$\tau = \frac{F_{\text{tot}} \cdot r_p}{\eta_{\text{drive}}} \quad \text{(belt-driven X and Y axes)}$$

$$\tau = \frac{F_{\text{tot}} \cdot L}{2\pi \cdot \eta_{\text{screw}}} \quad \text{(ball-screw-driven Z axis)}$$

where r_p is the pulley radius, L is the screw lead, and η is the transmission efficiency.

Gravity Torque Calculation (Z Axis)

For the ball-screw lift axis:

1. Moving mass: $m_Z = 11.00\,\text{kg}$
2. Gravity force: $F_{\text{grav}} = m_Z \cdot g = 11.00 \times 9.81 = 107.91\,\text{N}$
3. Acceleration force: $F_{\text{acc,Z}} = m_Z \cdot a_z = 11.00 \times 1.00 = 11.00\,\text{N}$
4. Total axial force: $F_{\text{tot,Z}} = 107.91 + 11.00 = 118.91\,\text{N}$
5. Screw torque:

$$T_{\text{screw}} = \frac{F_{\text{tot,Z}} \cdot \text{lead}}{2\pi \cdot \eta_{\text{screw}}} = \frac{118.91 \times 0.010}{6.283 \times 0.90} \approx 0.210\,\text{N·m}$$

6. Motor torque (direct drive): $T_{\text{motor,Z}} \approx 0.210\,\text{N·m}$
7. Screw speed: $v_z/\text{lead} = 0.20/0.010 = 20.0\,\text{rev/s} = 1200\,\text{RPM}$
8. Angular speed: $\omega = 2\pi \cdot 20.0 = 125.66\,\text{rad/s}$
9. Mechanical power: $P_{\text{motor,Z}} = T_{\text{motor,Z}} \cdot \omega \approx 26.4\,\text{W}$
10. With safety factor $SF_Z = 6$: $T_{\text{design,Z}} = 0.210 \times 6 = 1.26\,\text{N·m}$
11. Design-equivalent power: $P_{\text{design,Z}} = 1.26 \times 125.66 \approx 158.5\,\text{W}$

Thus, a servo in the 200–400 W range is appropriate for the Z axis.

Inertial Torque from Trajectory Motion (X Axis)

For the belt-driven X axis:

1. Acceleration force: $F_{\text{acc,X}} = m_X \cdot a_{xy} = 17.00 \times 1.00 = 17.00\,\text{N}$
2. Friction force: $F_{\text{fric,X}} = \mu \cdot m_X \cdot g = 0.02 \times (17.00 \times 9.81) = 3.34\,\text{N}$
3. Total force: $F_{\text{tot,X}} = 17.00 + 3.34 = 20.34\,\text{N}$
4. Pulley torque: $T_{\text{pulley,X}} = F_{\text{tot,X}} \cdot r_p = 20.34 \times 0.020 = 0.407\,\text{N·m}$
5. Motor torque: $T_{\text{motor,X}} = 0.407/0.90 \approx 0.452\,\text{N·m}$
6. Speed: $v_{xy}/(2\pi r_p) = 0.50/0.1257 \approx 3.98\,\text{rev/s} = 238.7\,\text{RPM}$
7. Angular speed: $\omega = 25.0\,\text{rad/s}$
8. Mechanical power: $P_{\text{motor,X}} = 0.452 \times 25.0 \approx 11.3\,\text{W}$
9. With safety factor $SF_X = 3$: $T_{\text{peak,X}} = 0.452 \times 3 = 1.36\,\text{N·m}$

Inertial Torque from Trajectory Motion (Y Axis)

For the belt-driven Y axis:

1. Acceleration force: $F_{\text{acc,Y}} = m_Y \cdot a_{xy} = 15.00 \times 1.00 = 15.00\,\text{N}$
2. Friction force: $F_{\text{fric,Y}} = 0.02 \times (15.00 \times 9.81) = 2.94\,\text{N}$
3. Total force: $F_{\text{tot,Y}} = 15.00 + 2.94 = 17.94\,\text{N}$
4. Pulley torque: $T_{\text{pulley,Y}} = 17.94 \times 0.020 = 0.359\,\text{N·m}$
5. Motor torque: $T_{\text{motor,Y}} = 0.359/0.90 \approx 0.399\,\text{N·m}$
6. Speed: same as X axis, 238.7 RPM, $\omega = 25.0\,\text{rad/s}$
7. Mechanical power: $P_{\text{motor,Y}} = 0.399 \times 25.0 \approx 9.97\,\text{W}$
8. With safety factor $SF_Y = 3$: $T_{\text{peak,Y}} = 0.399 \times 3 = 1.20\,\text{N·m}$

Payload and External Forces

The payload mass is included in the equivalent moving mass of each axis. No additional external process forces are considered for pick-and-place operation.

Assumptions

- Payload: $m_{\text{payload}} = 5.00\,\text{kg}$
- Extra assembly masses: $m_z = 6.00\,\text{kg}$ (Z carriage + screw + gripper), $m_y = 4.00\,\text{kg}$ (Y carriage & structure), $m_x = 2.00\,\text{kg}$ (X extras)
- Linear maximum speeds: $v_{xy} = 0.50\,\text{m/s}$ (X & Y), $v_z = 0.20\,\text{m/s}$ (Z)
- Accelerations: $a_{xy} = 1.00\,\text{m/s}^2$, $a_z = 1.00\,\text{m/s}^2$
- Pulley effective radius: $r_p = 0.020\,\text{m}$ (20 mm)
- Ball-screw lead: lead $= 0.010\,\text{m/rev}$ (10 mm per rev)
- Screw efficiency: $\eta_{\text{screw}} = 0.90$
- Drive efficiency (belts/gear): $\eta_{\text{drive}} = 0.90$
- Linear guide friction coefficient: $\mu = 0.02$
- Gravity: $g = 9.81\,\text{m/s}^2$
- Safety factors: $SF_{XY} = 3.0$ for X/Y peaks; $SF_Z = 6.0$ for Z (holding + safety)
- Workspace: $X = 1.000\,\text{m}, Y = 0.610\,\text{m}, Z = 0.410\,\text{m}$

Notation & Formulas

$$F_{\text{acc}} = m \cdot a, \quad F_{\text{fric}} = \mu \cdot m \cdot g, \quad F_{\text{tot}} = F_{\text{acc}} + F_{\text{fric}}$$

$$T_{\text{pulley}} = F_{\text{tot}} \cdot r_p, \quad T_{\text{motor}} = \frac{T_{\text{pulley}}}{\eta_{\text{drive}}}$$

$$T_{\text{screw}} = \frac{F_{\text{tot}} \cdot \text{lead}}{2\pi \cdot \eta_{\text{screw}}}$$

$$\text{rev/s} = \frac{v}{2\pi r_p}, \quad \text{RPM} = \text{rev/s} \cdot 60, \quad \omega = 2\pi \cdot \text{rev/s}$$

$$P = T \cdot \omega, \quad T_{\text{design}} = T_{\text{motor}} \cdot SF$$

Step-by-Step Numeric Calculations

Masses each axis must move:

$$m_Z = m_{\text{payload}} + m_z = 5.00 + 6.00 = 11.00\,\text{kg}$$

$$m_Y = m_{\text{payload}} + m_z + m_y = 5.00 + 6.00 + 4.00 = 15.00\,\text{kg}$$

$$m_X = m_{\text{payload}} + m_z + m_y + m_x = 5.00 + 6.00 + 4.00 + 2.00 = 17.00\,\text{kg}$$

Maximum Required Joint Torque Determination

From Sections 4.3 and 4.4, the maximum continuous torque requirements are:

$$T_{\text{motor,X}} = 0.452\,\text{N·m}, \quad T_{\text{motor,Y}} = 0.399\,\text{N·m}, \quad T_{\text{motor,Z}} = 0.210\,\text{N·m}$$

Safety Factor for Actuator Sizing

$$T_{\text{design}} = T_{\text{max}} \cdot SF$$

- For X and Y axis, $SF = 3$
- For Z axis, $SF = 6$

$$T_{\text{design,X}} = 1.36\,\text{N·m}, \quad T_{\text{design,Y}} = 1.20\,\text{N·m}, \quad T_{\text{design,Z}} = 1.26\,\text{N·m}$$

Torque Tables and Plots

Table 1: Torque requirements and selected motors

Axis	Continuous Torque (N·m)	Design Torque (N·m)	Selected Motor
X	0.45	1.36	750 W servo
Y	0.40	1.20	400 W servo
Z	0.21	1.26	750 W servo

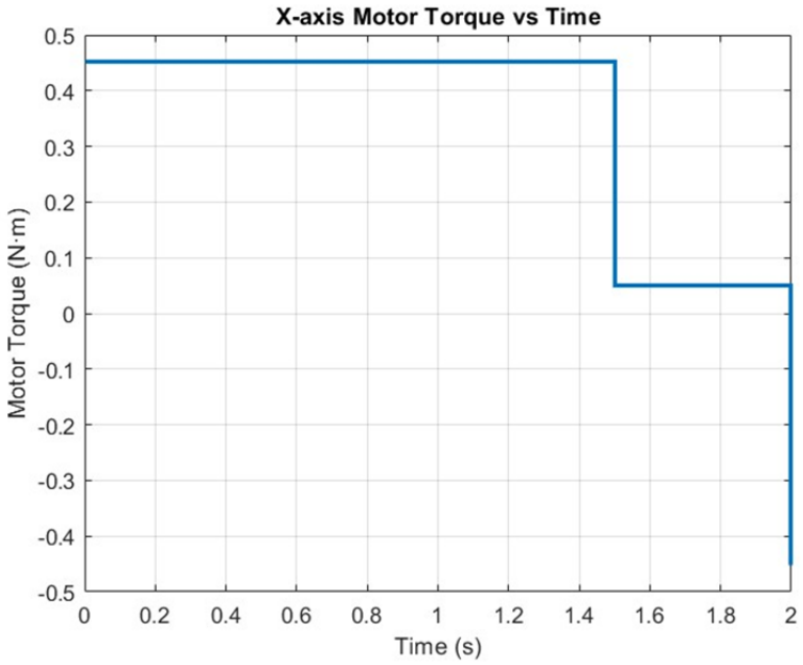

Figure 7: Torque vs. Time

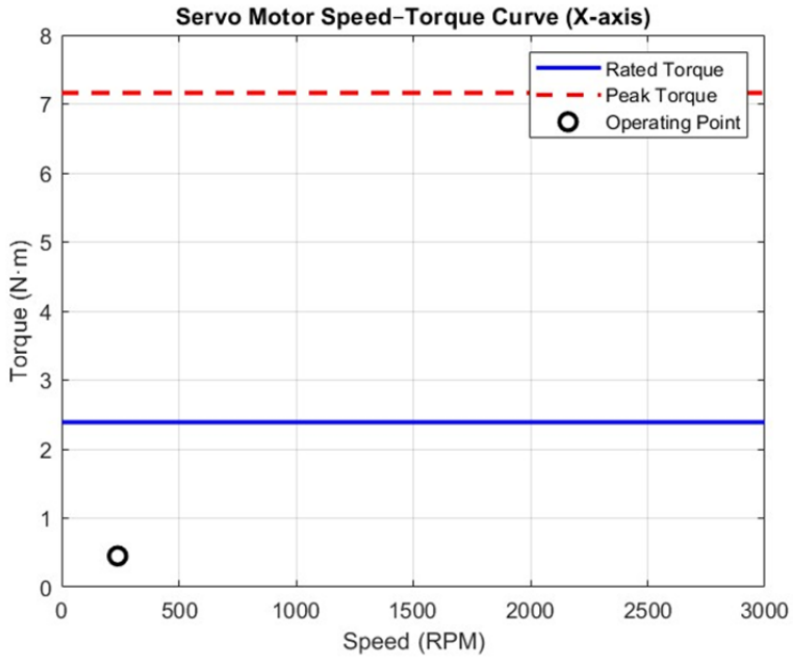

Figure 8: Speed-Torque curve.

Components & Design

Main Controller

- Siemens S7-1200 PLC
- Model: CPU 1212C AC/DC/Relay (6ES7212-1HE40-0XB0)

- 8 DI, 6 DO, expandable
- Reliable for industrial environment
- Expansion for Stepper/Servo Control (PTO Outputs)
- Siemens Signal Board SB 1231 (if extra AI required)
- Siemens Pulse-train module not required → CPU1212C already supports PTO up to 100 kHz

Servo Motors + Drivers (PPP Axes)

Payload 5 kg + 1 m horizontal travel requires servo motors, not steppers.

- **X Axis (1 m travel)** Servo motor: Delta ECMA-C20807RS (750 W, 3000 rpm) Driver: Delta ASDA-B2 series (Model: ASD-B2-0721-B) Transmission: HTD 5M belt, 25 mm wide OR ballscrew SFU1610
- **Y Axis (1 m travel)** Servo motor: Delta ECMA-C20604RS (400 W) Driver: Delta ASD-B2-0421-B (400 W driver) Transmission: HTD 5M belt, 20–25 mm width
- **Z Axis (0.8–1 m vertical travel)** Servo motor: Delta ECMA-C20807RS (750 W) Driver: Delta ASD-B2-0721-B Transmission: Ballscrew SFU1610 with BK/BF supports + linear guides

Linear Motion Components

- Linear Guides: Hiwin HGR20 rails (20 mm width)
- Blocks: HGW20CC (4 per axis)
- Ballscrews: SFU1610 (16 mm dia, 10 mm pitch) for Z; SFU1605 or HTD 5M belt drive for X & Y

Sensors for Material Classification

- **Metal Detection**: Inductive sensor LJ12A3-4-Z/BX, PNP Normally Open, 10–30 VDC
- **Plastic vs Paper Detection**: Capacitive sensor LJC18A3-B-Z/BX, PNP Normally Open, 10–30 VDC
- **Weight Measurement**: Load cell 5 kg aluminum plate, amplifier Waveshare 0–10 V output, connected to PLC AI0

Gripper (Pneumatic)

- Parallel Pneumatic Gripper: SMC MHZ2-20D
- Solenoid Valve: SMC SY3120-5LZD-01, 24 VDC, 3-way single solenoid
- Pneumatic Air Preparation: SMC AF20-N02 (filter), SMC AR20-N02 (regulator)

Power Supply System

- For PLC, sensors, valves (24 VDC bus): Meanwell LRS-350-24, 24 V, 14.6 A
- For servo drives (48 VDC bus): Meanwell RSP-750-48, 48 V, 60 A
- For load cell amplifier: same 24 V bus (accepts 12–24 V)

Safety Components

- Main Breaker: Schneider GV2-ME10 (adjustable 6–10 A)
- Emergency Stop: Schneider XALK178 mushroom E-stop
- Fuses: 5×20 mm fuse holders, 3A fuse for 24V logic, 10–15A fuse for servo power input
- Surge & Noise Protection: Ferrite cores on all servo cables, EMI filter Schaffner FN2070-10-06

Cables & Wiring

- Servo Motor Power Cables: 4-core shielded, 2.5 mm^2 (750 W), 1.5 mm^2 (400 W), length 8–10 m per axis
- Encoder/Signal Cables: 6-core shielded twisted pair, 0.5–0.75 mm^2, 10 m per axis
- Sensor Cables: 3-wire shielded, 0.5 mm^2, 5–10 m
- PLC Wiring: Single core 0.75 mm^2 for DO/DI and 24V distribution

Electrical Enclosure Components

- Metal enclosure 600×400×200 mm (steel)
- DIN rails (2× 350 mm)
- Cable ducts
- 24VDC terminal blocks (Weidmüller)
- Grounding bar
- Cooling fan 120 mm + filter

Wiring Diagram

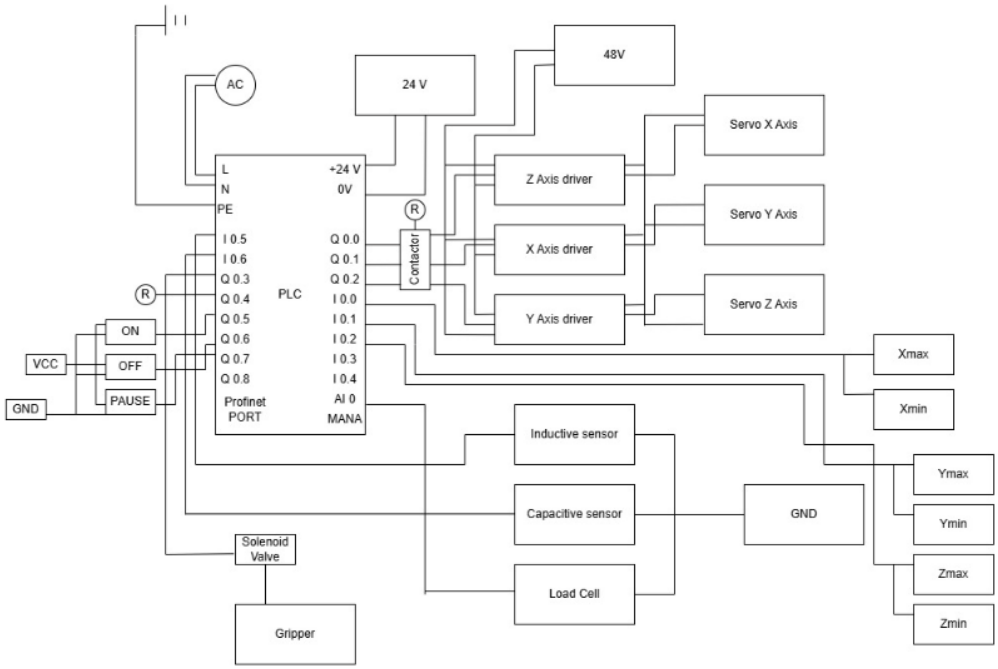

Figure 9: Wiring Diagram.

PLC Configuration

DI Tags

- I0.0 X_MINMAX
- I0.1 Y_MINMAX
- I0.2 Z_MINMAX
- I0.5 METAL_IND
- I0.6 CAP_HIGH

DO Tags

- Q0.0 SERVO_ENABLE X
- Q0.1 SERVO_ENABLE Y
- Q0.2 SERVO_ENABLE Z
- Q0.3 GRIPPER_OC
- Q0.4 COIL_CONTACTOR

- Q0.5 TOWER_RED
- Q0.6 TOWER_YELLOW
- Q0.7 TOWER_GREEN

AI Tag

- AI0 WEIGHT_mA (scale 4–20 mA → 0..FULL_SCALE grams)

PROFINET Drive Tags

DriveX

- DriveX.ControlWord
- DriveX.TargetPosition
- DriveX.TargetVelocity
- DriveX.StatusWord
- DriveX.ActualPosition
- DriveX.ErrorCode

DriveY

- DriveY.ControlWord
- DriveY.TargetPosition
- DriveY.TargetVelocity
- DriveY.StatusWord
- DriveY.ActualPosition
- DriveY.ErrorCode

DriveZ

- DriveZ.ControlWord
- DriveZ.TargetPosition
- DriveZ.TargetVelocity
- DriveZ.StatusWord
- DriveZ.ActualPosition
- DriveZ.ErrorCode

PLC Logic Chart

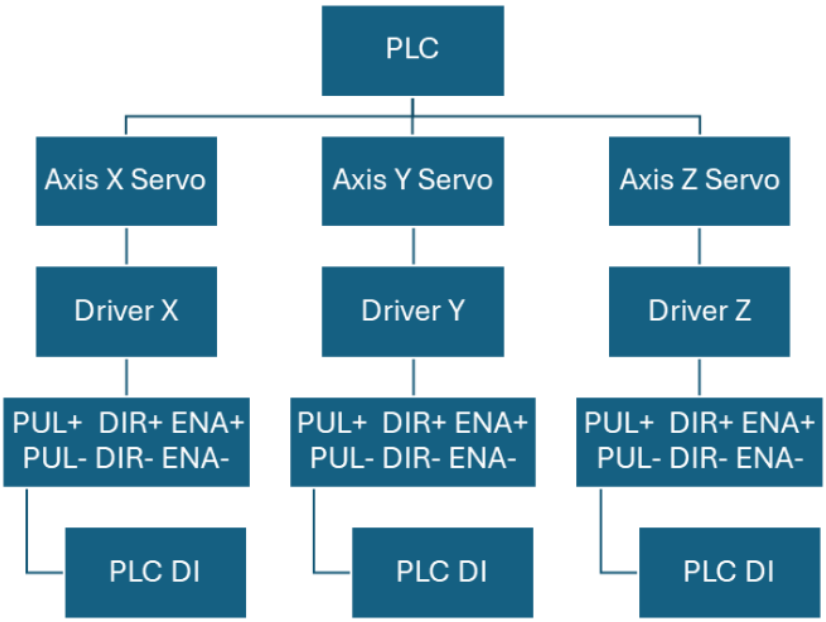

Figure 10: PLC Logic Chart

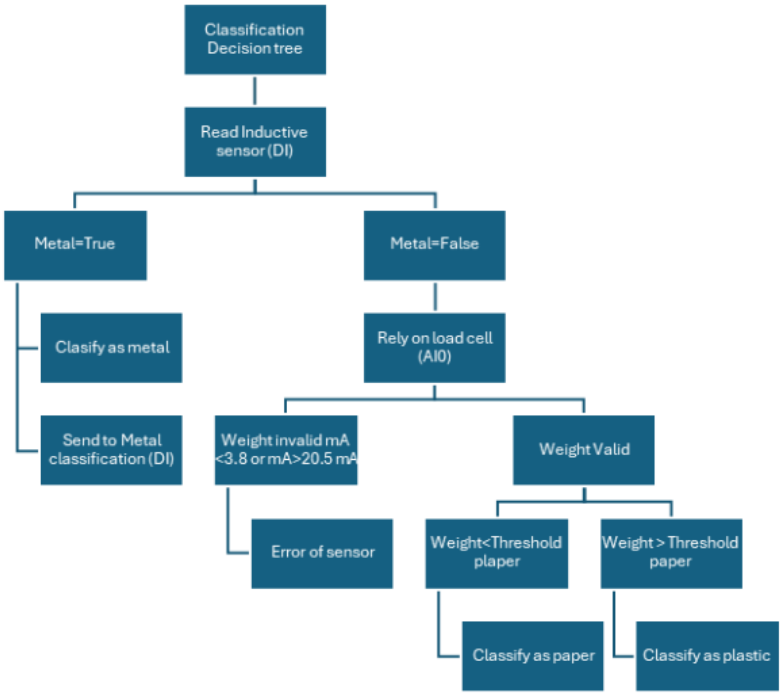

Figure 11: PLC Logic Chart

Part 5: Trajectory Generation Techniques

Project Overview

A PPP robot was implemented in MATLAB and Simulink. The folder `PPP_Robot_GUI` contains the key files:

- `ppp_fk.m`: forward kinematics
- `plot_ppp.m`: visualization
- `animate_pick_and_place.m`: joint-space trajectory example
- `ppp_dynamics.m`: simplified dynamics
- `gui_ppp_sim.m`: real-time control GUI

Joint-Space Trajectories

Interpolated trajectories in q_1, q_2, q_3 were implemented in `animate_pick_and_place.m`, producing non-linear Cartesian paths. **MATLAB animate_pick_and_place.m**

```
function T = ppp_fk(q)
    % Cinem tica directa para robot PPP tipo gr a
    % q = [q1; q2; q3]     desplazamientos prism ticos

    x = q(2); % eje Y movi ndose en X
    y = q(1); % eje Z movi ndose en Y
    z = q(3); % eje X movi ndose en Z

    T = [eye(3), [x; y; z]; 0 0 0 1];
end
```

Listing 1: MATLAB ppp_fk.m

Task-space trajectories

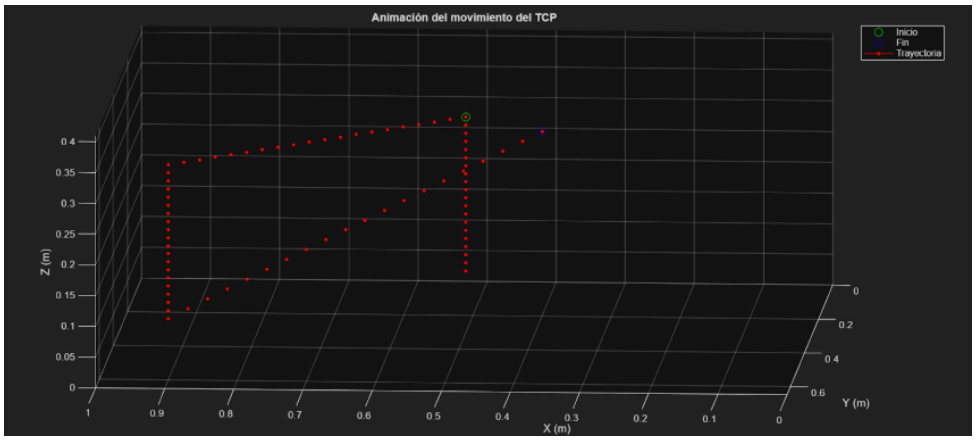

Figure 12: Task-space trajectories

Real-Time Trajectory Generation

The GUI allows real-time modification of q_1, q_2, q_3, demonstrating interactive trajectory generation.

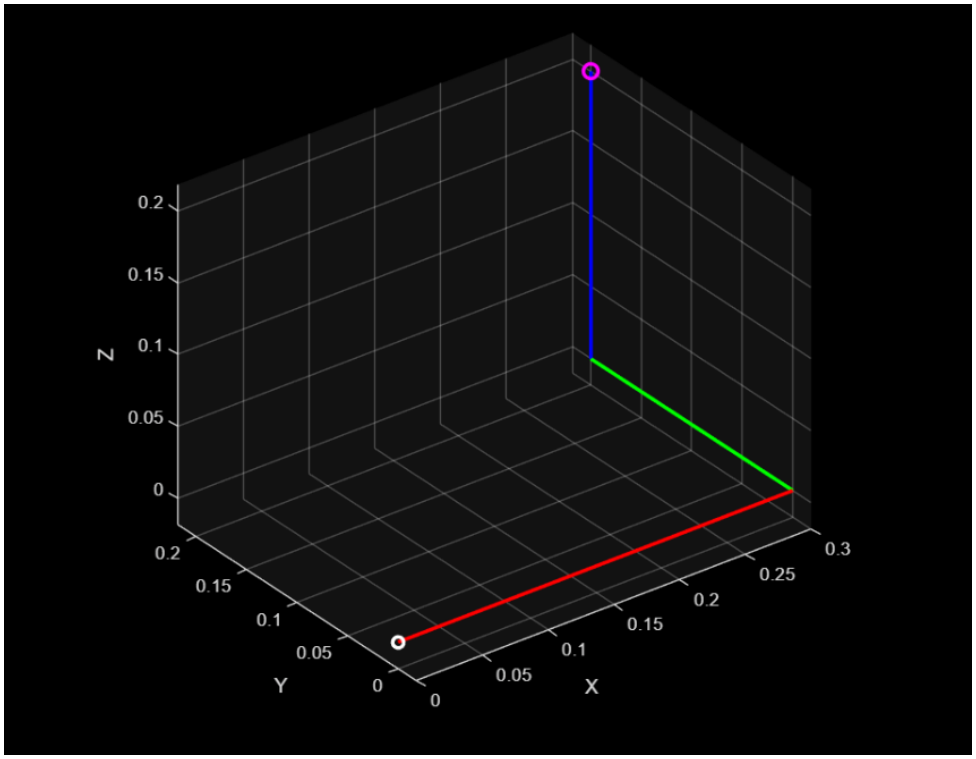

Figure 13: Task-space trajectories

```
function gui_ppp_sim()
    f = figure('Name','Simulador Robot
       PPP','Color','k','Position',[100 100 800 600]);

    % Sliders para q1, q2, q3
    uicontrol(f,'Style','text','String','q1 (Y vertical)',...
        'Position',[20 550 100
           20],'ForegroundColor','w','BackgroundColor','k');
    s1 =
       uicontrol(f,'Style','slider','Min',0,'Max',1,'Value',0.2,...
        'Position',[130 550 200 20],'Callback',@update_plot);

    uicontrol(f,'Style','text','String','q2 (X horizontal)',...
        'Position',[20 520 100
           20],'ForegroundColor','w','BackgroundColor','k');
    s2 =
       uicontrol(f,'Style','slider','Min',0,'Max',1,'Value',0.5,...
        'Position',[130 520 200 20],'Callback',@update_plot);

    uicontrol(f,'Style','text','String','q3 (Z profundidad)',...
        'Position',[20 490 100
           20],'ForegroundColor','w','BackgroundColor','k');
    s3 =
       uicontrol(f,'Style','slider','Min',0,'Max',1,'Value',0.3,...
        'Position',[130 490 200 20],'Callback',@update_plot);

    % Bot n de animaci n
    uicontrol(f,'Style','pushbutton','String','Animar Pick &
       Place',...
        'Position',[600 550 150 30],'Callback',@(src,event)
           animate_pick_and_place());

    % Ejes de visualizaci n
    ax = axes('Parent',f,'Position',[0.4 0.1 0.55 0.75]);
    set(ax,'Color','k');
    view(3); axis equal; grid on;
    xlabel('X'); ylabel('Y'); zlabel('Z');

    % Funci n de actualizaci n
    function update_plot(~,~)
        q = [s1.Value; s2.Value; s3.Value];
        axes(ax);
        plot_ppp(q, 'c');
    end

    update_plot();
end
```

Listing 2: MATLAB gui_ppp_sim.m

```
function plot_ppp(q, color)
    % Visualiza el robot PPP en configuraci n q
    T = ppp_fk(q);
    pos = T(1:3,4);

    clf; hold on; axis equal;
    set(gcf,'Color','k'); % fondo negro
    xlabel('X'); ylabel('Y'); zlabel('Z');
    view(3); grid on;

    % Ejes base
    plot3(0,0,0,'wo','MarkerSize',6,'LineWidth',2);
    % Trayectoria de cada eje
    plot3([0 q(2)], [0 0], [0 0], 'r', 'LineWidth', 2); % eje Y
          X
    plot3([q(2) q(2)], [0 q(1)], [0 0], 'g', 'LineWidth', 2); %
       eje Z     Y
    plot3([q(2) q(2)], [q(1) q(1)], [0 q(3)], 'b', 'LineWidth',
       2); % eje X     Z

    % End-effector
    plot3(pos(1), pos(2), pos(3), 'mo', 'MarkerSize', 8,
       'LineWidth', 2);
end
```

Listing 3: MATLAB plot_ppp.m

Task-Space Trajectories

Using `ppp_fk.m`, Cartesian paths (x, y, z) are mapped into joint space for linear or curved workspace paths.

```
function T = ppp_fk(q)
    % Cinem tica directa para robot PPP tipo gr a
    % q = [q1; q2; q3]     desplazamientos prism ticos

    x = q(2); % eje Y movi ndose en X
    y = q(1); % eje Z movi ndose en Y
    z = q(3); % eje X movi ndose en Z

    T = [eye(3), [x; y; z]; 0 0 0 1];
end
```

Listing 4: MATLAB ppp_fk.m

Trajectory Constraints

Velocity, acceleration, and workspace limits can be enforced during interpolation.

```
function tau = ppp_dynamics(q, dq, ddq, m)
    % Modelo din mico simplificado
    g = 9.81;
    M = diag(m); % matriz de inercia
    G = [m(1)*g; 0; 0]; % gravedad solo en eje vertical
    C = zeros(3,1); % sin Coriolis

    tau = M * ddq + C + G;
end
```

Listing 5: MATLAB ppp_dynamics.m

Discussion of Methods

- Joint-space: simple, but produces curved Cartesian paths.
- Task-space: intuitive, but requires IK.
- Real-time: ideal for testing and interactive control.

Part 6: Conclusion

Overall System Evaluation

The Automated Recycling Classifier (ARC) system successfully demonstrates the feasibility of using a low-cost PPP Cartesian robot combined with non-vision sensors for material classification in a benchtop recycling application. The integration of inductive sensing, capacitive dielectric measurement, and load-cell-based mass evaluation provides a sensor fusion approach capable of distinguishing metal, plastic, and paper with acceptable reliability under controlled conditions. The use of a fixed sensing station proved critical in reducing variability caused by distance, grounding effects, vibration, and dynamic loading. This design choice resulted in improved repeatability and signal consistency across classification cycles. The PPP gantry configuration offered mechanical simplicity, predictable kinematics, and ease of control, aligning well with the project objectives of affordability, modularity, and ease of implementation. While the system is not intended for industrial-scale throughput or complex material mixtures, it effectively fulfills its role as a compact automated sorting demonstrator and educational prototype.

Recommendations

- Implement signal filtering and averaging techniques to reduce sensor noise and improve capacitive and load cell measurement stability.
- Introduce calibration routines for the capacitive sensor and load cell to compensate for environmental variations such as humidity and temperature.
- Improve mechanical isolation of the weighing platform to minimize vibration-induced measurement errors.
- Explore adaptive thresholding or simple machine learning classifiers to enhance plastic–paper differentiation.
- Consider modular mechanical design upgrades to allow easy scaling of workspace dimensions or payload capacity.

Summary of Results

- Reliable execution of autonomous pick–classify–place cycles within the defined workspace was achieved.
- Metal objects were consistently detected using the inductive proximity sensor.

- Plastic and paper materials were differentiated using a combination of dielectric sensing and mass measurement.
- Stable and repeatable classification was demonstrated at a fixed sensing station.
- Effective integration of low-cost, non-vision sensors was achieved without compromising system functionality.

Suitability for Intended Application

The ARC platform is well suited for its intended application as a benchtop automated recycling classifier and proof-of-concept system. It addresses limitations associated with manual sorting by improving consistency, reducing human involvement, and enabling repeatable material classification using affordable hardware. Although the system is not designed for high-speed industrial recycling lines or complex composite materials, it is highly appropriate for educational and research environments, small-scale recycling demonstrations, and prototype development or sensor evaluation platforms. Its low cost, compact footprint, and straightforward control architecture make it a practical solution for controlled sorting scenarios.

Final Design Justification

The final ARC system design is justified by its strong alignment with the project objectives of affordability, simplicity, and functional effectiveness. The use of a PPP Cartesian manipulator ensures predictable motion, ease of simulation, and low mechanical complexity. Employing non-vision sensors significantly reduces system cost and computational requirements while maintaining acceptable classification performance under controlled conditions. The fixed sensing station design addresses key challenges related to measurement repeatability and sensor stability, enabling reliable sensor fusion. Overall, the final design represents a well-balanced compromise between performance, cost, and technical complexity, making it a suitable and defensible solution for automated recycling classification at the prototype level.