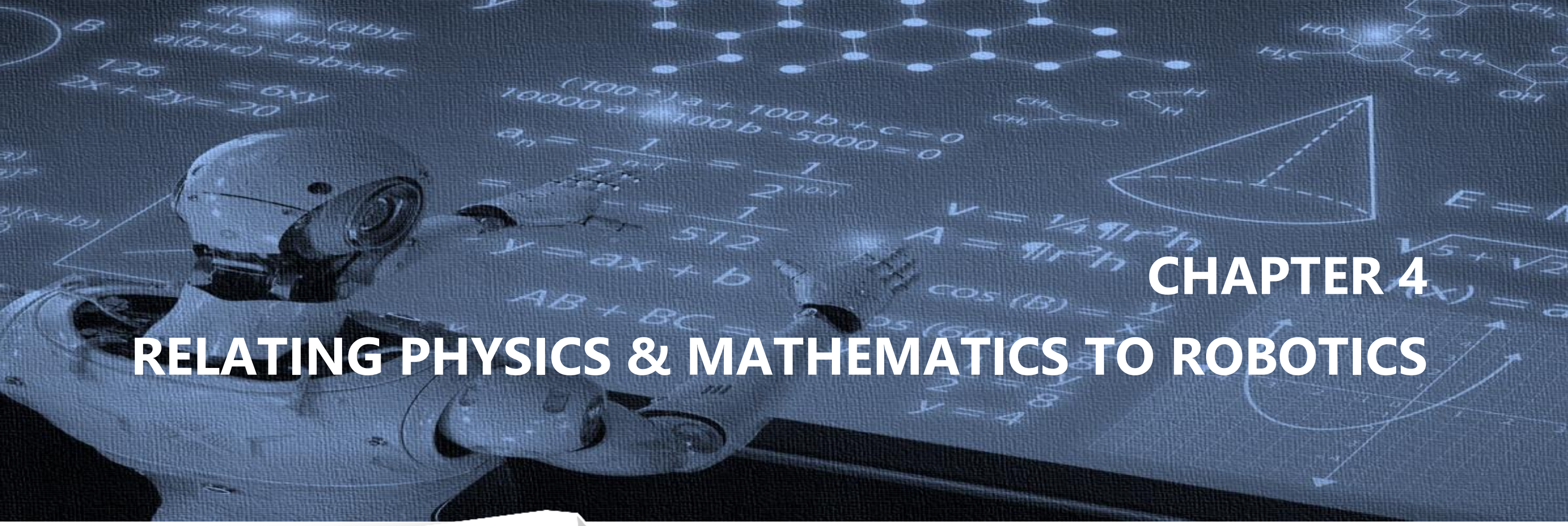

CHAPTER 4

RELATING PHYSICS & MATHEMATICS TO ROBOTICS

Learning Outcomes

- Principles of Motion, Force
- Introduction to Algorithm
- Matrix Operations
- Vector & Vector Spaces

Robotics relies on physics and mathematics to design, build, and control robots. Physics helps understand motion, forces, electricity, and mechanics, which are essential for movement, stability, and sensor operations. Mathematics is used for programming, calculating angles, distances, and controlling speed through algebra, trigonometry, and calculus. Geometry helps in designing robotic arms, while statistics support AI-based decision-making. By combining physics and mathematics, engineers create smarter robots for tasks like factory automation, space exploration, and self-driving cars. This chapter explores how these subjects play a crucial role in real-world robotics.

How Physics is used in Robotics

Physics plays a major role in robotics, helping engineers design and control robots effectively. It explains how robots move, sense their surroundings, and interact with the world.

- **Movement and Forces** – Newton's laws explain how robots move, while kinematics helps calculate speed and direction. Torque helps control robotic arms and joints.
- **Electricity and Magnetism** – Robots use electric motors, batteries, and sensors, all working with electrical and magnetic principles. DC motors and stepper motors use electromagnetism for movement, while magnetic sensors help in navigation.
- **Energy and Power** – Robots need energy sources like batteries or solar panels. Understanding how energy is stored and used helps improve battery life and efficiency.
- **Control and Feedback** – Robots use sensors like gyroscopes and cameras to adjust movement. Control systems like PID controllers help keep robots stable and precise.

- **Friction and Materials** – Friction affects how robots move, grip objects, and function on different surfaces. Choosing the right materials makes robots lighter, stronger, and more durable.
- **Sound and Vibration** – Robots use microphones and sensors to detect sounds. Understanding vibrations helps in smooth operation and reducing noise.
- **Vision and Optics** – Cameras and lasers help robots see, detect objects, and measure distances. LIDAR technology allows precise mapping and navigation.

Physics helps robots move, sense, and perform tasks accurately. Whether in industries, healthcare, space, or self-driving cars, physics is the key to building smarter robots.

How Mathematics is used in Robotics

Mathematics plays a key role in robotics, helping robots move, make decisions, and perform tasks accurately. Different math concepts are used to control robots and improve their performance.

- **Geometry and Trigonometry** – These help robots understand shapes, angles, and distances. They are used in designing robot parts and calculating movements, like how a robotic arm should bend or how a drone should turn.
- **Algebra and Equations** – Robots use algebra to calculate speed, direction, and position. It helps them plan paths and move efficiently. For example, robots solving mazes use algebra to find the shortest route.
- **Calculus and Motion Planning** – Calculus helps robots move smoothly by calculating speed, acceleration, and distance. It is important for self-driving cars and robotic arms to avoid jerky movements.
- **Probability and Statistics** – Robots make smart decisions using probability. For example, a self-driving car predicts if a pedestrian will cross the road. AI and machine learning also use probability to recognize patterns.
- **Matrix Algebra and Transformations** – Robots work in a 3D world, and matrices help them move and rotate correctly. This is used in robotic arms, computer vision, and virtual reality.
- **Graph Theory and Path finding** – Robots use maps to find the best way to reach a destination while avoiding obstacles. GPS systems and delivery robots use this to navigate.
- **Control Systems and Feedback Loops** – Robots need stable movements, and control systems adjust their speed and balance. For example, a self-balancing robot or a drone uses this to stay upright.
- **Fourier Analysis and Signal Processing** – Robots process sound and images by filtering unwanted noise. This helps in speech recognition and quality control in factories.
- **Artificial Intelligence and Neural Networks** – AI-powered robots use math to learn from data and make smart decisions. Concepts like deep learning and optimization improve their accuracy and efficiency.

Motion and Forces in Robotics

Motion and forces are important in robotics because they help robots move, stay balanced, and perform tasks properly. Engineers use these concepts to design robots that work smoothly and efficiently.

1. Newton’s Laws of Motion

- **First Law (Inertia):** A robot will stay still or keep moving in the same direction unless something pushes or stops it. This helps control how robots start, stop, or change direction.
- **Second Law (F = ma):** A heavier robot needs more force to move. This is why big robots need stronger motors.
- **Third Law (Action-Reaction):** When a robot pushes against a surface, the surface pushes back. This helps robots walk or pick up objects without slipping.

2. Types of Robot Movements

- **Straight-line motion:** Robots move in a straight path, like delivery robots.
- **Rotating motion:** Robot arms and wheels spin around a point.
- **Back-and-forth motion:** Some robots, like drills or vibrating sensors, move in a repeated pattern.

3. Friction and Stability

- Friction helps robots grip surfaces and avoid slipping. Too much friction slows them down, while too little makes control difficult.
- A robot’s **center of gravity** (balance point) affects stability. A low center of gravity prevents falling, especially for walking robots.

4. Forces That Help Robots Move

- **Torque:** The twisting force that moves wheels, arms, and joints.
- **Tension & Compression:** Robot parts feel pulling (tension) and pushing (compression) forces, which affect strength and durability.
- **Impact Forces:** Robots that jump or land need shock-absorbing materials to avoid damage.

5. Power and Energy

- Robots use electricity, hydraulics, or air pressure to move.
- Energy efficiency helps robots work longer without needing frequent charging.

By using these principles, engineers create robots that move properly, stay balanced, and perform tasks safely in industries, hospitals, and homes.

Algorithm for Programming and Problem-Solving

An **algorithm** is a step-by-step method to solve a problem or complete a task. In programming, an algorithm helps break down complex problems into smaller, manageable steps that a computer can follow. Writing a good algorithm is important for making programs efficient and error-free. To solve a problem using an algorithm, follow these steps:

1. Understanding the Problem
2. Designing the Algorithm

3. Testing the Algorithm
4. Optimizing the Algorithm

Ex: Algorithm: Finding the Largest of Three Numbers

```
Start

Take three numbers as input (A, B, and C).

Compare A with B and C.

If A is greater than both, A is the largest.

else, compare B with C.

If B is greater, B is the largest; otherwise, C is the largest.

Display the largest number.

Stop
```

How Matrix Operations Help Robots Move and Understand Space

In robotics, **matrices** are like **mathematical tools** that help robots move, turn, and understand their position in space. Robots need to perform precise actions, like **picking up objects, rotating arms, or navigating through an area**, and matrices make these calculations simple and efficient.

1. Importance of Matrices in Robotics

Robots work in a **3D space**, where they need to **move, rotate, and change positions** while understanding their surroundings. Since these movements involve complex calculations, **matrices** provide a structured and easy way to handle them.

Here are some important ways matrices are used in robotics:

□ **Positioning and Orientation (Homogeneous Transformations):** Helps robots know **where they are and which direction they are facing**.

□ **Movement in Space (Rotation and Translation)**: Used when a robot **moves forward, backward, up, down, or rotates in any direction**.

□ **Converting Between Different Positions (Coordinate Transformations):** Helps when a robotic arm or sensor needs to **understand its position in a different reference frame** (for example, a gripper's position relative to the robot's base).

□ **Defining Motion in Robotic Arms (Kinematic Equations)**: Used to calculate **how each joint in a robotic arm should move to reach a target**.

□ **Controlling Speed and Force (Jacobian Matrices):** Helps in adjusting the **speed and force** applied by robotic arms, ensuring smooth and safe movements.

2. Transformations Using Matrices

2.1. Homogeneous Transformation Matrices

A **homogeneous transformation matrix** represents both **rotation** and **translation** in a single 4×4 matrix. It allows us to describe how a point or object moves from one coordinate frame to another.

A general **homogeneous transformation matrix** is:

$$T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$

where:

- **R** is a **3×3 rotation matrix**.
- **t** is a **3×1 translation vector**.
- **0** is a row vector **(0 0 0)** to maintain homogeneity.
- **1** is a scalar to maintain transformations in homogeneous coordinates.

This matrix helps define how an object or robotic arm moves in space by combining rotation and translation in one operation.

2.2. Rotation Matrices

Rotating a robotic arm or object requires **rotation matrices**, which represent rotations about the **X, Y, and Z axes**. These matrices are derived using trigonometric functions **(sine and cosine)**.

Rotation about the X-Axis

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

Rotation about the Y-Axis

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

Rotation about the Z-Axis

$$R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Each matrix rotates a point or object **counter clockwise** by **θ degrees** about the respective axis. To perform **3D rotations**, these matrices are **multiplied together** in a specific order.

2.3. Translation Matrices

A **translation matrix** moves a point or object from one position to another in space. It is represented as:

$$T = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where (x, y, z) represents the translation along each axis.

Applying a translation matrix moves a point **(P)** from one position to another:

$$P' = T \cdot P$$

where P' is the new position after translation.

3. Handling Coordinate Systems in Robotic Arms

Robotic arms consist of multiple **links** and **joints**, each with its own **coordinate frame**. The movement of one joint affects the overall position of the end-effector. To describe these **relative movements**, we use:

1. **Denavit-Hartenberg (D-H) Parameters** – A systematic way of assigning coordinate frames to each robotic link.
2. **Transformation Matrices** – To transition from one coordinate frame to another.

Each joint in a robotic arm has its transformation matrix **T_i**, and the final position of the end-effector is found by multiplying all individual transformations:

$$T_{final} = T_1 \cdot T_2 \cdot T_3 \cdots T_n$$

where n is the number of joints.

4. Rotations and Translations in Robotic Arms

For a **robotic manipulator**, performing a **pick-and-place** task requires:

1. **Calculating the target position** in world coordinates.
2. **Applying inverse kinematics** to determine the required joint angles.
3. **Using transformation matrices** to determine how the end-effector should rotate and move.

A **6-DOF robotic arm** uses a **sequence of rotations and translations** to reach a desired point in space. For example, to move an object from **position A** to **position B**:

- Use **translation matrices** to move it along X, Y, or Z.
- Use **rotation matrices** to adjust orientation.

- Multiply the matrices in the correct order.

Moving a Robotic Arm

Assume a robotic arm needs to **rotate by 45° around the Z-axis** and then **translate 10 cm along the X-axis**.

1. **Rotation Matrix (Z-axis, 45°)**

$$R_z(45^\circ) = \begin{bmatrix} 0.707 & -0.707 & 0 & 0 \\ 0.707 & 0.707 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2. **Translation Matrix (X = 10 cm)**

$$T_x(10) = \begin{bmatrix} 1 & 0 & 0 & 10 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3. **Final Transformation Matrix**

$$T_{final} = T_x(10) \cdot R_z(45^\circ)$$

Multiplying these matrices gives the new position and orientation of the robotic arm.

Vectors

A **vector** is a mathematical concept that has **both size and direction**. It is different from a **scalar**, which only has size (like temperature or speed). Vectors help describe movement, force, and position in space, making them very important in robotics. A vector can be shown as an **arrow**, where the **length** represents how big the movement or force is, and the **direction** shows where it is pointing. Mathematically, vectors are written as **(x, y, z)**, where **x** is left/right movement, **y** is up/down movement, and **z** is forward/backward movement.

Robots use vectors for many tasks. In **navigation**, vectors help robots decide how to move from one place to another, like self-driving cars following the best route. In **motion control**, robotic arms use vectors to apply just the right amount of force to pick up or hold objects without dropping or crushing them. Vectors also help with **rotation and orientation**, such as helping drones stay balanced while flying or robotic arms move at the correct angles. In **artificial intelligence (AI)**, vectors are used in pattern recognition, such as facial recognition, where a face is converted into a set of numbers for identification.

Vectors can also be **added** and **scaled**. If a robot moves from **(3,4,0)** to **(2,-1,0)**, its final position is found by adding the numbers together, giving **(5,3,0)**. If a vector is **multiplied by a**

number (scalar), its size increases or decreases. This is useful for adjusting speed, force, or distance in robotic movements.

What is a Vector Space?

A **vector space** is a collection of objects called **vectors** that can be **added** together and **multiplied by numbers** (called scalars). These vectors follow some specific rules to form a proper mathematical structure. **Vector spaces** are widely used for **calculating direction** in various fields. Here's how:

1. Physics (Motion and Forces)

- Vectors help in representing **velocity, acceleration, and force**.
- The **direction** of motion or force is determined using vector operations.
- Example: The movement of an object in space is given by a vector that shows both **speed** (magnitude) and **direction**.

2. Navigation and GPS

- GPS systems use vector spaces to calculate the shortest path and direction between locations.
- Example: Airplanes and ships navigate using **vector calculations** to stay on course.

3. Computer Graphics & Game Development

- Vectors are used in **3D modelling** to define the direction of movement, lighting, and object positioning.
- Example: In video games, a car's movement in a racing game is controlled by **vectors** to ensure realistic turns and motion.

4. Robotics & Artificial Intelligence

- Robots calculate their movement and orientation using vectors.
- Example: A robotic arm moves in a specific direction using vector transformations.

5. Engineering & Architecture

- Used in structural analysis to determine how forces act on buildings and bridges.
- Example: Engineers calculate **stress and strain** in materials using vectors.

Vectors in Force Calculation

Force is a physical quantity that has both **magnitude** (strength) and **direction**. Since it has direction, it is represented as a **vector**.

Representing Force as a Vector

A force vector is written as:

$$\mathbf{F} = (F_x, F_y, F_z)$$

where Fx,Fy,FzF_x, F_y, F_z are the components of force in the **x, y, and z directions**. For example, if a force of **10 N** is applied at an angle of **30°** to the horizontal, its components are:

- **Horizontal force**:

$$F_x = F \cos 30^\circ$$

- **Vertical force:**

$$F_y = F \sin 30^\circ$$

Using $F = 10$ N:

$$F_x = 10 \cos 30^\circ = 8.66 \text{ N}$$

$$F_y = 10 \sin 30^\circ = 5 \text{ N}$$

So, the force vector is $(8.66, 5)$.

Vectors for Position in Space

A **position vector** tells us the exact location of a point in space relative to a fixed starting point, called the **origin**. In three-dimensional space, it is written as **(x, y, z)**, where **x, y,** and **z** are the coordinates of the point. For example, if a point is at **(3, 4, 5)**, it means it is **3 steps along the x-axis, 4 steps along the y-axis, and 5 steps along the z-axis** from the origin.

To find the **distance** between two points, we use a special formula based on the Pythagorean theorem. If two points **A(x_1, y_1, z_1)** and **B(x_2, y_2, z_2)** are given, their distance is found using:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

For example, the distance between points A(1, 2, 3) and B(4, 6, 8) is **7.07 units**. The **direction** of a position vector tells us where the point is located relative to the origin. We can find this direction using a **unit vector**, which is the position vector divided by its magnitude (length). This helps in understanding **where** an object is, not just how far it is.

Position vectors are very useful in **physics, navigation, robotics, and computer graphics**. In robotics, they help locate a robotic arm's exact position. In navigation, GPS systems use them to find places accurately. In video games, position vectors decide where a character or object should appear on the screen.

Position vectors are very useful in **tasks that involve many directions** or dimensions. In **three-dimensional space**, a position vector **(x, y, z)** helps us find the exact location of an object. But in advanced fields like **robotics, navigation, and computer graphics**, we sometimes work with even more dimensions, such as time or extra movements.

For example, in **robotics**, position vectors help a robotic arm move in different directions smoothly. In **self-driving cars**, they help calculate the exact position of the car in real-time. In **video games and animations**, position vectors decide where objects should be placed and how they should move in a 3D world. Since these tasks involve movement in many directions, position vectors are very important for making everything work accurately and smoothly.

Chapter Highlights

Physics in Robotics

- **Motion & Forces** – Newton's laws, torque, and stability.
- **Electricity & Magnetism** – Motors, sensors, and batteries.
- **Energy & Power** – Efficient energy use.
- **Control & Feedback** – Sensors for stability.
- **Friction & Materials** – Grip, movement, and durability.
- **Sound & Vibration** – Noise reduction and smooth operation.
- **Vision & Optics** – Cameras and LIDAR for navigation.

Mathematics in Robotics

- **Geometry & Algebra** – Movement calculations and path planning.
- **Calculus & Probability** – Smooth motion and AI decision-making.
- **Matrix Algebra & Graph Theory** – 3D movement and navigation.
- **Control Systems & Fourier Analysis** – Stability and signal processing.
- **AI & Neural Networks** – Learning and adaptation.

Motion & Forces in Robotics

- **Newton's Laws** – Motion and balance.
- **Types of Motion** – Linear, rotational, and repetitive.
- **Forces & Stability** – Torque, friction, and energy efficiency.

Exercise

Multiple-Choice Questions (MCQs)

1. Which of the following is NOT a major role of physics in robotics?
   a) Vision and optics
   b) Sound and vibration
   c) Financial modelling
   d) Friction and materials
2. Newton's first law of motion states that:
   a) Every action has an equal and opposite reaction
   b) A heavier robot needs more force to move
   c) An object stays in motion unless acted upon by an external force
   d) Robots can store and use energy efficiently
3. What mathematical concept helps robots find the shortest path in a maze?
   a) Calculus
   b) Algebra
   c) Trigonometry
   d) Graph Theory

4. Which force is responsible for rotating robotic arms and wheels?
   a) Tension
   b) Torque
   c) Compression
   d) Friction
5. What role does probability play in robotics?
   a) Helps robots navigate using LIDAR
   b) Helps robots make smart decisions
   c) Helps in designing robot arms
   d) Helps reduce energy consumption
6. Which of the following is an example of rotating motion in robotics?
   a) A delivery robot moving in a straight path
   b) A robotic arm picking up an object
   c) A vibrating robotic sensor
   d) A robot using AI for decision-making
7. What is the main purpose of a PID controller in robotics?
   a) Improve energy efficiency
   b) Control movement and stability
   c) Reduce friction in robotic joints
   d) Optimize deep learning algorithms
8. What is the function of matrices in robotics?
   a) Enhancing AI algorithms
   b) Helping in motion planning and rotation
   c) Reducing energy consumption
   d) Improving robotic sound detection
9. Which of the following helps in reducing noise in robotics?
   a) Fourier Analysis
   b) Graph Theory
   c) Newton’s Laws
   d) Trigonometry
10. What is the key advantage of using a low center of gravity in a robot?
   a) Increases movement speed
   b) Reduces energy consumption
   c) Improves balance and stability
   d) Enhances AI decision-making

True or False Questions

1. Robots do not use electromagnetism for movement.
2. Friction always makes robotic movement smoother.
3. AI-powered robots use probability to make smart decisions.
4. Newton’s second law states that a robot needs more force to move if it is heavier.
5. Graph Theory is not used in robotics path finding.

Fill in the Blanks

1. _______ is the twisting force that helps robotic arms and wheels move.
2. _______ and _______ help robots detect objects and measure distances.

3. Newton's _______ law explains that for every action, there is an equal and opposite reaction.
4. _______ helps robots process sound and images by filtering unwanted noise.
5. AI-powered robots use _______ to recognize patterns and make decisions.

Assertion and Reasoning Questions

1. **Assertion:** Robots need a stable center of gravity for proper movement.
   Reason: A low center of gravity improves a robot's balance and prevents falling.
   a) Both A and R are true, and R explains A
   b) Both A and R are true, but R does not explain A
   c) A is true, but R is false
   d) A is false, but R is true
2. **Assertion:** Robots use sensors to detect their surroundings.
   Reason: Sensors like gyroscopes and cameras help in feedback control.
   a) Both A and R are true, and R explains A
   b) Both A and R are true, but R does not explain A
   c) A is true, but R is false
   d) A is false, but R is true
3. **Assertion:** Newton's second law states that an object will remain at rest unless acted upon by an external force.
   Reason: This law helps in calculating the force needed to move a robot.
   a) Both A and R are true, and R explains A
   b) Both A and R are true, but R does not explain A
   c) A is false, but R is true
   d) Both A and R are false
4. **Assertion:** Graph Theory is used in robotic navigation.
   Reason: It helps robots in path finding and avoiding obstacles.
   a) Both A and R are true, and R explains A
   b) Both A and R are true, but R does not explain A
   c) A is true, but R is false
   d) A is false, but R is true
5. **Assertion:** Robots can use calculus for motion planning.
   Reason: Calculus helps robots move smoothly by calculating speed and acceleration.
   a) Both A and R are true, and R explains A
   b) Both A and R are true, but R does not explain A
   c) A is true, but R is false
   d) A is false, but R is true

Short Answer Questions

1. How does Newton's third law apply to robotics?
2. What is the role of friction in robotic movement?
3. Why do robots use matrix algebra?
4. What is the importance of probability in AI-powered robots?
5. How do control systems like PID controllers help in robotics?

Long Answer Questions

1. Explain how different types of motion (straight-line, rotating, back-and-forth) are used in robotics with examples.
2. Describe the role of AI and machine learning in robotics. How do they help in decision-making?
3. Discuss the importance of Newton's laws in robotics, giving real-world applications.
4. How does mathematics help in robot navigation? Explain the role of trigonometry, graph theory, and path finding.
5. What are the different types of forces acting on a robot? Explain with examples.