

Introduction to Robotics: Lecture 2 Notes

Based on Lectures by Professor Oussama Khatib

January 26, 2025

Contents

1	Introduction to Kinematics	1
2	Robotic Manipulators: Links and Joints	1
2.1	Types of Joints	1
2.2	Degrees of Freedom	1
2.2.1	Calculating DOF	1
2.3	Configuration Representation	2
2.3.1	Generalized Coordinates	2
2.3.2	Joint Space	2
2.3.3	Operational Space	2
2.4	Redundancy	2
3	Spatial Descriptions and Transformations	3
3.1	Points and Vectors	3
3.2	Coordinate Frames	3
3.2.1	Transformation Between Frames	3
3.3	Rotation Matrices	3
3.3.1	Definition	3
3.3.2	Components of Rotation Matrix	3
3.3.3	Inverse Rotation	3
3.3.4	Example	4
3.4	Homogeneous Transformations	4
3.4.1	Definition	4
3.4.2	Transforming Points	4
3.4.3	Composition of Transformations	4
3.4.4	Inverse Transformation	4
3.5	Representing Positions and Orientations	4
3.5.1	Euler Angles	4
3.5.2	Axis-Angle Representation	5
3.5.3	Quaternions	5
3.5.4	Rotation Matrices	5
4	Transformation Interpretations	5
4.1	Mapping vs. Operators	5
4.2	Transformation Operators	5
4.2.1	Rotation Operators	5
4.2.2	Translation Operators	5
4.2.3	Combined Rotation and Translation	5
5	Inverse Transformations	6
5.1	Inverse Rotation	6
5.2	Inverse Homogeneous Transformation	6

6	Serial Kinematic Chains	6
6.1	Denavit-Hartenberg (DH) Convention	6
6.2	Forward Kinematics	6
6.3	Building the Kinematic Model	6
7	Examples and Applications	6
7.1	Planar Manipulator	6
7.1.1	Redundant Manipulator	7
7.2	Transformation Between Links	7
7.2.1	Rotation about X-axis	7
7.2.2	Translation along Z-axis	7
7.2.3	Combined Transformation	7
8	Conclusion	7

1 Introduction to Kinematics

Kinematics is the study of motion without considering the forces that cause it. It is fundamental in robotics for understanding and modeling the positions, orientations, and motions of robotic manipulators. This lecture focuses on the foundational concepts of kinematics, including spatial descriptions, transformations, and representations of position and orientation.

2 Robotic Manipulators: Links and Joints

A **robotic manipulator** consists of a series of **links** connected by **joints**. The structure can be divided into:

- **Base Link:** The fixed link, serving as the reference frame.
- **Moving Links:** Intermediate links connected through joints.
- **End-Effector:** The last link, used to interact with the environment (e.g., gripper, tool).

The primary goal is to control the position and orientation of the end-effector to perform tasks such as manipulation.

2.1 Types of Joints

There are two fundamental types of joints in robotic manipulators:

1. **Revolute Joint (Rotational):** Allows rotation about a fixed axis.
2. **Prismatic Joint (Translational):** Allows translation along a fixed axis.

Both joint types provide **one degree of freedom (DOF)**.

Higher DOF Joints Complex joints like **spherical joints** (3 DOF) can be modeled by combining multiple revolute joints with zero link lengths.

2.2 Degrees of Freedom

The **degree of freedom** refers to an independent parameter that defines the configuration of a system.

2.2.1 Calculating DOF

For a manipulator with n moving links:

- Each link in space requires 6 parameters (3 for position, 3 for orientation).
- Total parameters without constraints: $6n$.
- Each joint introduces constraints, allowing only 1 DOF (introducing 5 constraints).
- Total constraints from joints: $5n$.
- Therefore, total DOF: $6n - 5n = n$.

For a manipulator with a fixed base, the total DOF is equal to the number of joints, n .

Mobile Bases For robots with mobile bases (e.g., humanoid robots), the base contributes additional DOF (typically 6). Thus, the total DOF becomes $n + 6$.

2.3 Configuration Representation

Configuration Parameters Parameters that fully describe the position and orientation of each link in the manipulator. They can be:

- **Redundant:** More parameters than necessary.
- **Minimal:** The smallest set of parameters that fully describe the system.

2.3.1 Generalized Coordinates

A minimal set of independent parameters that uniquely define the configuration of the system. The number of generalized coordinates corresponds to the system's DOF.

Example If each link is represented using three points (vectors), each with three components, this results in 9 parameters per link. However, these parameters are not all independent due to geometric constraints. Generalized coordinates aim to minimize the number of parameters by exploiting these dependencies.

2.3.2 Joint Space

Also known as the **configuration space**, it is defined by the joint variables (e.g., joint angles for revolute joints). A point in joint space represents a specific configuration of the manipulator:

$$\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_n]^T$$

Importance in Motion Planning The joint space is crucial for planning movements, avoiding obstacles, and ensuring the manipulator operates within its physical limits.

2.3.3 Operational Space

Also known as the **task space**, defined by the position and orientation of the end-effector:

$$\mathbf{x} = [x, y, z, \phi, \theta, \psi]^T$$

These are the **operational coordinates** or **task coordinates**, representing the end-effector's pose in space.

Degrees of Freedom of the End-Effector

- In 3D space: Maximum of 6 DOF (3 for position, 3 for orientation).
- In a plane: 3 DOF (2 for position, 1 for orientation).

2.4 Redundancy

A manipulator is **redundant** if it has more DOF than required to perform a task:

$$\text{Degree of Redundancy} = n - m_0$$

Where:

- n is the number of manipulator DOF.
- m_0 is the number of DOF required for the end-effector.

Benefits of Redundancy

- Improved obstacle avoidance.
- Enhanced dexterity and flexibility.
- Ability to optimize secondary criteria (e.g., energy efficiency, singularity avoidance).

3 Spatial Descriptions and Transformations

3.1 Points and Vectors

A point P in space is represented by a position vector \mathbf{p} from a reference origin O . The vector depends on the choice of origin; changing the origin changes the vector components.

3.2 Coordinate Frames

A coordinate frame consists of an origin and three orthogonal unit vectors $\{\mathbf{x}, \mathbf{y}, \mathbf{z}\}$. Frames are used to describe the position and orientation of objects.

- **Reference Frame (Frame A):** The fixed frame.
- **Body Frame (Frame B):** Attached to a moving rigid body (e.g., a link).

3.2.1 Transformation Between Frames

To describe Frame B relative to Frame A, we need:

- **Rotation Matrix R_B^A :** Describes the orientation of Frame B relative to Frame A.
- **Position Vector \mathbf{d}_B^A :** Describes the origin of Frame B relative to Frame A.

3.3 Rotation Matrices

3.3.1 Definition

A rotation matrix $R \in SO(3)$ represents the orientation of one frame relative to another:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

Properties:

- Orthogonal: $R^T R = R R^T = I$
- Determinant: $\det(R) = +1$

3.3.2 Components of Rotation Matrix

The columns of R_B^A are the components of Frame B's axes expressed in Frame A:

$$R_B^A = [\mathbf{x}_B^A \quad \mathbf{y}_B^A \quad \mathbf{z}_B^A]$$

Computing Components Each component is found by projecting the axes of Frame B onto the axes of Frame A (e.g., $\mathbf{x}_B^A = [r_{11}, r_{21}, r_{31}]^T$).

3.3.3 Inverse Rotation

The inverse of a rotation matrix is its transpose:

$$(R_B^A)^{-1} = (R_B^A)^T = R_A^B$$

3.3.4 Example

Consider a frame rotated around the x -axis:

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$$

3.4 Homogeneous Transformations

3.4.1 Definition

A homogeneous transformation matrix combines rotation and translation into a single 4×4 matrix:

$$T = \begin{bmatrix} R & \mathbf{d} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

Where:

- R is the 3×3 rotation matrix.
- \mathbf{d} is the 3×1 translation vector.

3.4.2 Transforming Points

To transform a point from Frame B to Frame A:

$$\begin{bmatrix} \mathbf{p}^A \\ 1 \end{bmatrix} = T_B^A \begin{bmatrix} \mathbf{p}^B \\ 1 \end{bmatrix}$$

Example Given T_B^A and a point \mathbf{p}^B , compute \mathbf{p}^A by matrix multiplication.

3.4.3 Composition of Transformations

Transformations can be composed via matrix multiplication:

$$T_C^A = T_B^A T_C^B$$

3.4.4 Inverse Transformation

The inverse of a homogeneous transformation:

$$T^{-1} = \begin{bmatrix} R^T & -R^T \mathbf{d} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

3.5 Representing Positions and Orientations

3.5.1 Euler Angles

Orientation can be represented using three angles, specifying a sequence of rotations about axes. Common sequences include:

- Intrinsic rotations (body-fixed axes): Z-X-Z, Z-Y-X.
- Extrinsic rotations (space-fixed axes): X-Y-Z, Y-Z-X.

Advantages and Limitations

- Simple interpretation.
- May suffer from singularities (gimbal lock).

3.5.2 Axis-Angle Representation

Any rotation can be represented by an axis \mathbf{k} and an angle θ :

$$R = I + \sin \theta [\mathbf{k}]_{\times} + (1 - \cos \theta) [\mathbf{k}]_{\times}^2$$

Where $[\mathbf{k}]_{\times}$ is the skew-symmetric matrix of \mathbf{k} .

3.5.3 Quaternions

Quaternions provide a singularity-free representation of orientation using four parameters:

$$q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}, \quad \|q\| = 1$$

Used extensively in 3D graphics and aerospace applications.

3.5.4 Rotation Matrices

Alternatively, orientations can be represented directly using rotation matrices, providing a full description with nine parameters (though only three are independent).

4 Transformation Interpretations

4.1 Mapping vs. Operators

Mapping Changing the **description** of the same vector from one frame to another without altering the vector itself.

Operators Applying a transformation to a vector, thereby changing the vector's magnitude or direction (e.g., rotating or translating a point in space).

4.2 Transformation Operators

4.2.1 Rotation Operators

A rotation can be applied to a vector to produce a new vector:

$$\mathbf{p}_2 = R\mathbf{p}_1$$

Example Rotating a point \mathbf{p}_1 about the x -axis by angle θ .

4.2.2 Translation Operators

A translation moves a vector by a displacement \mathbf{d} :

$$\mathbf{p}_2 = \mathbf{p}_1 + \mathbf{d}$$

4.2.3 Combined Rotation and Translation

A general transformation combining rotation and translation:

$$\mathbf{p}_2 = R\mathbf{p}_1 + \mathbf{d}$$

5 Inverse Transformations

5.1 Inverse Rotation

As previously noted, the inverse of a rotation matrix is its transpose:

$$R^{-1} = R^T$$

5.2 Inverse Homogeneous Transformation

The inverse of a homogeneous transformation matrix is:

$$T^{-1} = \begin{bmatrix} R^T & -R^T \mathbf{d} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

Interpretation This inversion involves transposing the rotation matrix and transforming the translation vector appropriately.

6 Serial Kinematic Chains

6.1 Denavit-Hartenberg (DH) Convention

A systematic method to assign coordinate frames to each link and joint of a manipulator, using four parameters per link:

- **Link Length** (a_i): Distance between Z_{i-1} and Z_i along X_i .
- **Link Twist** (α_i): Angle between Z_{i-1} and Z_i about X_i .
- **Link Offset** (d_i): Distance from X_{i-1} to X_i along Z_{i-1} .
- **Joint Angle** (θ_i): Angle from X_{i-1} to X_i about Z_{i-1} .

6.2 Forward Kinematics

Computing the position and orientation of the end-effector given the joint parameters:

$$T_{0n} = \prod_{i=1}^n T_i$$

Where T_i is the homogeneous transformation from link $i - 1$ to link i .

6.3 Building the Kinematic Model

1. Assign coordinate frames to each link following the DH convention.
2. Determine the DH parameters for each link.
3. Write the homogeneous transformation matrices using these parameters.
4. Multiply the matrices to obtain the end-effector transformation.

7 Examples and Applications

7.1 Planar Manipulator

Consider a planar manipulator with three revolute joints:

- Joint variables: $\theta_1, \theta_2, \theta_3$.
- End-effector position: $[x, y]$.
- End-effector orientation: α .

Configuration and Operational Space

- **Joint Space (Configuration Space):** $\theta = [\theta_1, \theta_2, \theta_3]^T$.
- **Operational Space (Task Space):** $\mathbf{x} = [x, y, \alpha]^T$.

7.1.1 Redundant Manipulator

Adding an extra joint introduces redundancy:

- $n = 4$ joints.
- Operational DOF: $m_0 = 3$.
- Degree of redundancy: $n - m_0 = 1$.

Implications of Redundancy Multiple joint configurations can achieve the same end-effector pose, allowing for optimization of other criteria.

7.2 Transformation Between Links

Example Compute the homogeneous transformation between consecutive links using rotation and translation matrices.

7.2.1 Rotation about X-axis

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

7.2.2 Translation along Z-axis

$$T_z(d) = \begin{bmatrix} I & \begin{bmatrix} 0 \\ 0 \\ d \end{bmatrix} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

7.2.3 Combined Transformation

$$T = T_z(d)R_x(\theta)$$

8 Conclusion

Understanding spatial descriptions and transformations is fundamental in robotics kinematics. Mastery of coordinate frames, rotation matrices, and homogeneous transformations allows for precise modeling of manipulator positions and orientations. These foundational concepts are essential for designing control algorithms, motion planning, and ensuring successful interaction with the environment.