Introduction To Robotics

Adithya Nair

August 12, 2024

Contents

Part I

Robotics

Chapter 1

Unit 1

1.1 Introduction

This course is mainly going to focus on **Manipulators**. These machines are used to manipulate positions and the state of the objects in an environment. We're going to break down their movements into Dynamics Analysis akin to the work done in Computational Mechanics.

1.2 Syllabus

- Overview Of Robotics
- Kinematics Of Simple Robotic Systems
- Dynamics And Control Of Simple Robotic Systems

1.3 Glossary

1. Actuator - Does work upon receiving voltage
2. Encoder - Sensor that measures raw angle data.

Software for robotics - **v-rep**, MATLAB This software involves making a CAD model, and apply a mathematical model. Unity can also be used in making such models.

1.4 Degree Of Freedom

The degree of freedom of a mechanical system is defined as the no. of independent paramets need to completely define its position in space at a given time..

The degree of freedom is defined with respect to a reference frame. If the object is free to rotate and move, it means it has 6 degrees of freedom. Localization - Finding the position and orientation of an object in 3-dimensional space.

We call a system 'fully actuated' when there are as many actuators as there are degrees of freedom.

$$\text{No. of controlling inputs} < \text{No. of degrees of freedom}$$

Underactuated systems contain lesser actuators than the number of degrees of freedom

$$\text{No. of controlling inputs} = \text{No. of degrees of freedom}$$

Redundant systems contain more actuators than the number of degrees of freedom

$$\text{No. of controlling inputs} > \text{No. of degrees of freedom}$$

1.5 Kinematic Pair

Linkages are the basic elements of all mechanisms and robots. Links are rigid body member with nodes, and joints are connection between links at nodes. Allows relative motion between links.

1.6 Robotic Manipulator

- Why study kinematics and dynamics of robotic manipulator
  - To manipulate an object in space
  - Understand the workspace and limitations of a robotic manipulator
  - Understand and estimate contact force between end-effector and object being manipulated.

1.7 Pose of a rigid body

A rigid body is completely defined in space by its position and orientation with respect to a reference.

We use the terminology 'Inertial reference frame' to mean an observer where Newton's laws of physics apply. We use the terminology 'Inertial reference frame' to mean an observer where Newton's laws of physics apply and the frame itself does not accelerate. Generally the base of the robotic manipulator is treated as the inertial reference frame

We use unit vectors $\hat{x}, \hat{y}, \hat{z}$ to describe the basis vectors. For the orientation of the rigid body, since they lie in 3d space, we must define new basis vectors to define the orientation, $\hat{x'}, \hat{y'}, \hat{z'}$

$$\hat{x'} = x'_x\hat{x} + x'_y\hat{y} + x'_z\hat{z}$$

$$\hat{y'} = y'_x\hat{x} + y'_y\hat{y} + y'_z\hat{z}$$

$$\hat{z'} = z'_x\hat{x} + z'_y\hat{y} + z'_z\hat{z}$$

1. When the frame is translationally different from the original frame.

$$^{A}\vec{P} = {}^{B}\vec{P} + {}^{A}\vec{P}_{Borg}$$

   Where $^{A}\vec{P}$ is the position vector of P with respect to A, $^{B}\vec{P}$ is the position vector of P with respect to B, and $^{A}\vec{P}_{Borg}$ is the vector of A to B.

2. When the frame is oriented differently from the original frame. Generating a rotation matrix $^{B}_{A}R$ to rotate vectors from a frame B to A. Where A and B are reference frames, with B being oriented differently than A.

$$\begin{bmatrix} \hat{X}_b \cdot \hat{X}_a & \hat{Y}_b \cdot \hat{X}_a & \hat{Z}_b \cdot \hat{X}_a \\ \hat{X}_b \cdot \hat{Y}_a & \hat{Y}_b \cdot \hat{Y}_a & \hat{Z}_b \cdot \hat{Y}_a \\ \hat{X}_b \cdot \hat{Z}_a & \hat{Y}_b \cdot \hat{Z}_a & \hat{Z}_b \cdot \hat{Z}_a \end{bmatrix}$$

   We can simply the matrix into 3 column vectors, with the notation $^{A}X_{B}$ which means B with respect to A

$$\begin{bmatrix} ^{A}\hat{X}_B & ^{A}\hat{Y}_b & ^{A}\hat{Z}_B \end{bmatrix}$$

3. When the frame is both translationally and oriented different from the original frame

$$^A\vec{P} =^A_B R^B\vec{P} +^A \vec{P}_{Borg}$$

To simplify the equations, we write.

$$^A\vec{P} =^A_B T^B P$$

Where T becomes,

$$^A_BT = \begin{bmatrix} ^A_BR_{3\times3} & ^AP_{Borg} \\ 0_{1\times3}1_{1\times1} & \end{bmatrix}$$

This T is the homogeneous transformation matrix.

The Rotation Matrix belongs to a category of matrices called $SO(3)$

1.8 System Nomenclature

Definition 1.8.1

A dynamical system is a particle or ensemble of particles whose state varies over time and thus obeys differential equations involving time derivatives.

Definition 1.8.2

The state of a dynamic system is the smallest set of variables(called state variables) such that the knowledge of these variables at $t = t_0$, together with the knowledge of the input for $t \geq t_0$, completely determines the behaviour of the system for any time $t \geq t_0$. The number of state variables is the same as the number of inital conditions needed to completely solve the system models

Definition 1.8.3

If n state variables are needed to completely describe the behaviour of a given system, then the n state variables can be considered the components of a vector X. Such a vector is called a state vector.

Definition 1.8.4

The state space is the n -dimensional space whose coordinate axes consist of the x_1 axis, $x_2 axis$ where $x_1, \cdots, x_n$

Suppose you started observing your model at point t_0. You can read any point in the state space and immediately figure out exactly how the system was behaving at any given point.

Input Variables: $U_{n\times1} = [U_1, U_2, \cdots, U_r]^T$ State Variables: $X_{n\times1} = [X_1, X_2, \cdots, X_r]^T$ Output Variables: $Y_{n\times1} = [Y_1, Y_2, \cdots, Y_r]^T$

We can write the derivatives of the state variables as functions of the inputs and the state variables and time.

We can then write these functions as one vector.

$$\dot{X}(t) = F(U, x, t)$$

We can similarly write the derivatives of the outputs as functions of the inputs and state variables.

$$\dot{Y}(t) = G(U, X, t)$$

We write the states' derivative as a linear combination, which can then be written as a matrix.

$$\dot{X}(t) = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{1n} \\ \vdots & \ddots & & \vdots \\ a_{n1} & a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \begin{bmatrix} b_{11} & b_{21} & \cdots & b_{1n} \\ \vdots & \ddots & & \vdots \\ b_{n1} & b_{n1} & \cdots & b_{nn} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix}$$

We get the equation,

$$\dot{X}(t) = A_{n\times n}X_{n\times 1} + B_{n\times r}U_{r\times 1}$$

A is the 'state' matrix, and B is the 'input' matrix
Similarly we can write,

$$\dot{Y}(t) = C_{m\times n}X_{n\times 1} + D_{m\times r}U_{r\times 1}$$

We apply the Laplace transform to convert integrals and derivatives into algebraic operations.

$$U(t) \to G(s) \to Y(t)$$

We get the transfer function $G(s)$ by the formula

$$G(s) = \frac{Y(s)}{U(s)}$$

So we can get our outputs, by

$$Y(s) = G(s)U(s)$$

Definition 1.8.5

A block diagram of a system is a pictorial representation of the functions performed by each component and of the flow of signals. Such a diagram depicts the interrelationships that exist among the various components. Differing from a purely abstract mathematical representation, a block diagram has the advantage of indicating more realistically the signal flows of the actual system.

1.9 Modeling Of Dynamical Systems.