Introduction To Robotics

Adithya Nair

October 23, 2024

Contents

I Robotics 3

1 Unit 1 5
- 1.1 Introduction . . . 5
- 1.2 Syllabus . . . 5
- 1.3 Glossary . . . 5
- 1.4 Degree Of Freedom . . . 5
- 1.5 Kinematic Pair . . . 6
- 1.6 Robotic Manipulator . . . 6
- 1.7 Pose of a rigid body . . . 6
- 1.8 Denavit Hartenberg Parameters . . . 7
- 1.9 Rotation Matrix . . . 8
- 1.10 Derivation Using DH Parameters . . . 8
- 1.11 Skew Symmetric Matrix . . . 8

2 Unit 2 - Kinematics Of Robotic Manipulators 11
- 2.1 Forward And Inverse Kinematics . . . 11
- 2.2 Inverse Kinematics Of 2R Planar Structures . . . 11
- 2.3 Forward Kinematics And Inverse Kinematics Of 3R Planar Structures . . . 11
- 2.4 Jacobian Forward Kinematics . . . 12
- 2.5 Singularity . . . 12
- 2.6 Finding The Jacobian quickly . . . 13
- 2.7 Lab Questions . . . 13

3 Unit 3- Statics And Dynamics 15
- 3.1 Proof . . . 15

4 Velocity and force ellipse 17
- 4.1 Finding the locus of isotropic points . . . 17

5 Dynamics 18

II Practice Sheets **19**

III End-Sem Project **21**

IV Dynamics **23**

V Syllabus **24**

VI Project **26**

6 Robotics Project **27**
6.1 Forward Kinematics 27
6.2 Inverse Kinematics 27
6.3 Statics 27
6.4 Dynamics 27
6.5 Report Is Required 27

7 Modelling Project **28**
7.1 Modelling 28
7.2 Dynamical Response To Standard Input 28
7.3 Frequency Response 28

Part I

Robotics

Textbook

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, $\widehat{x'}, \widehat{y'}, \widehat{z'}$

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

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

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

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

$$^{A}_{B}T = \begin{bmatrix} ^{A}_{B}R_{3\times3} & ^{A}P_{Borg} \\ 0_{1\times3} & 1_{1\times1} \end{bmatrix}$$

This T is the homogeneous transformation matrix.

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

1.8 Denavit Hartenberg Parameters

These parameters are used to assign the same co-ordinate frames while dealing with robotic manipulators to ensure that the entire scientific community are working under the same conventions.

- Number the link sequentially from 0 to n. Giving us $n + 1$ links.
- Number the joints sequentially from 1 to n, Giving us n joints.
- Number of co-ordinate frames: $n + 1$
- The Z_i axis is aligned with the $(i + 1)^th$ joint axis.
- X_i is defined along the common normal between the Z_i and Z_{i-1} axis.(Common normal is the line which is perpendicular to both of these axes.)
- Y_i is obtained via cross product between Z_i and X_i axis.
- The origin is placed at this point of intersection of x_i, y_i and z_i
- We define H_i(which is the point of intersection of the common normal of the next axis with the current axis.), and 4 new terms:
  - a_i - Offset distance between two adjacent joint axes(The distance between O_i and H_{i-1})
  - d_i - Distance between H_{i-1} and O_{i-1}
  - α_i Angle between Z_i and Z_{i-1} when viewed from X_i, this is also known as the twist angle.
  - θ_i Angle between X_i and X_{i-1} when viewed from Z_i,

For a 3R Planar serial chain manipulator:

	$\mathrm{a_i}$	$\mathrm{d_i}$	α_i	θ_i
Joint i = 1	$\mathrm{L_1}$	0	0	θ_1
Joint i = 2	$\mathrm{L_2}$	0	0	θ_2
Joint i = 3	$\mathrm{L_3}$	0	0	θ_3

1.9 Rotation Matrix

We have three reference frames, with a common origin. With the notation we have setup we can say, $\{0\}, \{1\}, \{2\}$ can be defined for a point P

$$
\begin{aligned}
{}^0P &=^0_1 R^1P \\
{}^0P &=^0_2 R^2P \\
{}^1P &=^1_2 R^2P \\
{}^0_2R &=^0_1 R^1_2R
\end{aligned}
$$

1.10 Derivation Using DH Parameters

These 4 parameters can be expressed by,

$$
{}^{i-1}A_i = T(z,d)T(z,\Theta), T(x,\alpha), T(x,a)
$$

1.11 Skew Symmetric Matrix

A matrix such that $A = -A^T$

So this means that,

$$
S + S^T = 0
$$

$$
\vec{a} = a_x\hat{i} + \vec{a_y}\hat{j}\vec{a_z}\hat{k}
$$

We define a matrix function,

$$
S(\vec{a}) = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x0 & \end{bmatrix}
$$

So we can say for the basis vectors

$$
S(\hat{i}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}
$$

$$
S(\hat{j}) = \begin{bmatrix} 0 & 0 & 1 & \\ 0 & 0 & 0 & \\ -1 & 0 & 0dt & i0 \end{bmatrix}
$$

$$
S(\hat{k}) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}
$$

Multiplying the matrix function's output with a vector is the same as doing a cross product with those two vectors.

$$S(\vec{a})P = \vec{a} \times \vec{p}$$

This matrix operation also has linearity in operations.

$$S(\alpha\vec{a} + \beta\vec{b})P = \alpha S(\vec{a}) + \beta S(\vec{b})$$

When multiplied with a rotation matrix.

$${}^{a}_{b}RS(\vec{a}){}^{a}_{b}R^{T} = S({}^{a}_{b}R\vec{a})$$

We know that,

$$\begin{aligned} RR^T &= I \\ R(\theta)R^t(\theta) &= I \\ \text{Taking the derivative} & \\ \frac{dR(\theta)}{d\theta}R^T(\theta) + R(\theta)\frac{dR^T(\theta)}{d\theta} &= 0 \\ \frac{dR(\theta)}{d\theta}R^T(\theta) + R(\theta)\left(\frac{dR^T(\theta)}{d\theta}\right)^T &= 0 \end{aligned}$$

What this means is that the term $\frac{dR(\theta)}{d\theta}R^T(\theta)$ is a skew-symmetric matrix. This means that the the term can be written as S,

$$S = \frac{dR(\theta)}{d\theta}R^T(\theta)$$

In other words, S becomes an operator on R to give us the derivative.

$$S(\hat{k})R(\theta) = \frac{dR(\theta)}{d\theta}$$

Taking both sides with respect to time,

$$\frac{dR(\theta)}{d\theta}\frac{d\theta}{dt} = \dot{\theta}S(\hat{k})R(\theta)$$

In other words,

$$\dot{R} = S(\dot{\theta}\hat{k})R(\theta)$$

So we have,

$$\dot{R} = S(\vec{\omega})R(\theta)$$

$$S(\vec{\omega}) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ \omega_y & \omega_x & 0 \end{bmatrix}$$

$$\Omega = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ \omega_y & \omega_x & 0 \end{bmatrix}$$

$$\begin{gathered}
{}^0P =^0_1 R^1P \\
{}^0P = R^1P \\
{}^0\dot{P} = \dot{R}^1P + R^1\dot{P} \\
{}^0\dot{P} = S(\vec{\omega})R^1P \\
{}^0\dot{P} = \omega \times R^1\vec{p} \\
{}^0\dot{P} = \omega \times R^1\vec{p} \\
\vec{v} = \vec{\omega} \times \vec{r}
\end{gathered}$$

Chapter 2

Unit 2 - Kinematics Of Robotic Manipulators

2.1 Forward And Inverse Kinematics

Forward kinematics is the use of the joint space to get to the task space. Inverse Kinematics is the use of the task space to get to the joint space.

Workspace - The space of all points that the end effector can reach.

2.2 Inverse Kinematics Of 2R Planar Structures

We solve for q_2 and eliminate q_1

$2\mathrm{L}_1 L_2$

We have two cases,
Case 1: $-1 < \kappa < 1$ We have two distinct and real solutions of q_2
Case 2: $|\kappa| = 1$ We have one solution
Case 3: $|\kappa| < 1$ No solution exists

2.3 Forward Kinematics And Inverse Kinematics Of 3R Planar Structures

$$X_e = L_1 \cos q_1 + L_2 \cos(q_1 + q_2) + L_3 cos(q_1 + q_2 + q_3)$$

For a 3R Planar structure, the way to find the forward kinematics is straightforward.

What we get for the inverse kinematics is with 3 knowns, ${}^0X_e, {}^0Y_e, \phi$ where $(\phi = q_1 + q_2 + q_3)$ we reduce the problem down to ${}^0X_2, {}^0Y_2$ by finding,

$$^0X_e - L_3 \cos\phi = L_1 \cos q_1 + L_2 \cos q_{12}$$
$$^0Y_e - L_3 \sin\phi = L_1 \sin q_1 + L_2 \sin q_{12}$$

Now we know that those terms are nothing but

$$
\begin{aligned}
{}^0X_2 &= L_1 \cos q_1 + L_2 \cos q_{12} \\
{}^0Y_2 &= L_1 \sin q_1 + L_2 \sin q_{12}
\end{aligned}
$$

Which we can solve the same as the 2R Planar Structure

2.4 Jacobian Forward Kinematics

Given $\dot{q_1}, \dot{q_2}$, how do we get ${}^0X_e, {}^0Y_e$? and similarly how do we get the angular velocities of a machine from the end effector's velocity.

We know that for 2R Planar SCMs,

$$
\begin{aligned}
{}^0X_e &= L_1 c q_1 + L_2 c(q_1 + q_2) \\
{}^0Y_e &= L_1 s q_1 + L_2 s(q_1 + q_2) \\
{}^0\dot{X}_e &= -\dot{q_1} L_1 s q_1 - L_2 s(q_1 + q_2) \times (\dot{q_1} + \dot{q_2}) \\
{}^0\dot{Y}_e &= +\dot{q_1} L_1 c q_1 + L_2 c(q_1 + q_2) \times (\dot{q_1} + \dot{q_2})
\end{aligned}
$$

We have,

$$\dot{X}_{2\times1} = J_{2\times2} \times \dot{q}_{2\times1}$$

We now have this for a specific case, when we generalize,

$$\dot{X} = \begin{bmatrix} {}^0_n v \\ {}^0_n \omega \end{bmatrix} = J\dot{q}$$

So now we can write the term

$${}^0_n\omega = \sum_{i=1}^{n} \dot{\theta}_i$$

2.5 Singularity

So now we have,

$$
\begin{aligned}
\dot{X} &= J\dot{q} \\
\dot{J} &= q^{-1}X \\
J^{-1} &= \frac{adj(J)}{det(J)}
\end{aligned}
$$

We term singularity to be the case where the determinant of J is 0

$$det(J) = L_1 L_2 \sin\theta_2$$

This means that we're only depending on θ_2

Which cannot be $n\pi$

What this means is that when θ_2 should never be zero.

This means that when the arm is fully extended or retracted then it is not a point that the robot can recover from.

2.6 Finding The Jacobian quickly

$$\mathrm{J}_i = \begin{bmatrix} {}^{0}_{i-1}z \times {}^{i-1}_{n} p^* \\ {}^{0}_{i-1}z \end{bmatrix}$$

Where ${}^{i-1}_{n}P^*$ is the position vector of $\{n\}th$ origin relative to $\{i-1\}^{th}$ frame but expressed in $\{0\}^{th}$ frame.

2.7 Lab Questions

1. Question 1 Consider the 3R Planar SCM with $L_1 = 2m$, $L_2 = 3m, L_3 = 4m$

```
L = [2;3;4];

% Workspace Points
WS_points = zeros(0,0);
```

   (a) Forward Kinematics, Find the co-ordinates of the end-effector in the base frame:
      i. For $q = [q_1 q_2 q_3]^T = [30\deg 45\deg 60\deg]^T$
      ii. For $q = [q_1 q_2 q_3]^T = [270\deg 75\deg 10\deg]^T$
   (b) Inverse kinematics, Find the joint space variables,
      i. For $\phi = q_1 + q_2 + q_3 = 0$ and $[{}^0X_e, {}^0Y_e] = [2, 4]$
      ii. For $\phi = q_1 + q_2 + q_3 = 45$ and $[{}^0X_e, {}^0Y_e] = [5, 5]$
   (c) Workspace Analysis Draw the workspace of the 3R planar serial chain manipulator using forward and inverse kinematics.
      i. For $\phi = 0$ and
      ii. For $\phi = 45\deg$

      i. Using Forward Kinematics
      ii. Using Inverse Kinematics

$$\begin{bmatrix} {}^0X_e \\ {}^0Y_e \end{bmatrix} = \begin{bmatrix} L_1 cos q_1 + L_2 cos(q_{12}) + L_3 cos(\phi) \\ L_1 sin q_1 + L_2 sin(q_{12}) + L_3 sin(\phi) \end{bmatrix}$$

$$\begin{bmatrix} {}^0X_e - L_3 cos(\phi) \\ {}^0Y_e - L_3 sin(\phi) \end{bmatrix} = \begin{bmatrix} L_1 cos q_1 + L_2 cos(q_{12}) \\ L_1 sin q_1 + L_2 sin(q_{12}) \end{bmatrix}$$

$$\begin{bmatrix} {}^0X_2 \\ {}^0Y_2 \end{bmatrix} = \begin{bmatrix} L_1 cos q_1 + L_2 cos(q_{12}) \\ L_1 sin q_1 + L_2 sin(q_{12}) \end{bmatrix}$$

$${}^0X_2^2 +^0 Y_2^2 = (L_1^2) + (L_2)^2 + 2L_1L_2\cos(q_2)$$

$$q_2 = \cos^{-1}\kappa$$

We plot the workspace by checking the point $(L_1 + L_2 + L_3, 0)$ and $(-(L_1 + L_2 + L_3), 0)$ as they are the points at which the arm is fully extended. This means that the radius of the circle we need to check is just a circle where $r = (L_1 + L_2 + L_3)$

We then look at points where $\kappa > -1$ and $\kappa < 1$ since those are the points which are reachable in the workspace.

```
P = zeros(3,1);
P(3) = 0;
range = linspace(-(L(1) + L(2) + L(3)),L(1) + L(2) + L(3));
[X,Y] = ndgrid(range);

% Position vector of {2} from ground frame
for i = 1:length(X)
  for j = 1:length(Y)
    P(1) = X(i,j);
    P(2) = Y(i,j);
    X2 = [P(1) - L(3)*cos(P(3));P(2) - L(2)*sin(P(3))];
    kappa = (X2(1)^2 + X2(2)^2 - L(1)^2 - L(2)^2)/2*L(1)*L(2);
        if(kappa >= -1 && kappa <= 1)
        % Appending to the workspace
        WS_points = cat(1,WS_points,[P(1) P(2)])
        end
  end
end
```

Now we plot,

```
figure('1, "visible", "off", 'units', 'normalized','outerposition',[0 0 0.5 1])

grid on; grid minor
ylim([-((L(1) + L(2) + L(3))-1 L(1) + L(2) + L(3))+1 ])
xlim([-((L(1) + L(2) + L(3))-1 L(1) + L(2) + L(3))+1 ])
axis equal
xlabel('X-Axis(m)')
ylabel('Y-Axis(m)')

ans = "../images/octave-chart.png";
```

Chapter 3

Unit 3- Statics And Dynamics

Statics is the study of the forces and torques acting on bodies that are at rest or in a state of equilibrium.

How does this play into robotics? This comes in handy in situations where the robot must stand still. It needs to apply forces to counteract all the forces on it to stay stationary.

We know from inverse kinematics that,

$$\dot{X} = J\dot{q}$$

The final answer here is,

$$\tau = J^T F$$

Where τ is the joint torques and F is the End-Effector's wrench vector, which containeds both forces and torques.

We invert both sides

$$\dot{q} = J^{-1}\dot{X}$$

and

$$F = (J^T)^{-1}\tau$$

With singularity, we're unable to map the end effector's veloticy to joint space velocities. It also means we're unable to map joint torques(τ) to the Wrench vector(F) In the neighbourhood of sinularity, samll velocity in task space will cause very high velocity in joint space, and small joint torques will cause very high forces and torques.

3.1 Proof

Using work method,

Work done by external agent at the Ende Effector = Work done by actuators at the joint.

$$\vec{F} \cdot \partial\vec{X} = \tau \cdot \partial\theta$$

$$F^T \cdot \partial\vec{X} = \tau^T \cdot \partial\theta$$

From Jacobian relation:

$$F^T = J\partial\theta => \partial X = J\partial\theta$$

So,

$$F^T J d\theta = \tau^T d\theta$$

$$F^T J = \tau^T d$$

Taking the transpose on both sides

$$J^T F = \tau$$

Using Newton's second law, Define the wrench vector for the end effector

$$F_{6\times 1} = \begin{bmatrix} f_{3\times 1} \\ \\ n_{3\times 1} \end{bmatrix}$$

For a given link i

$$\vec{f} = 0$$

$$\begin{aligned} f_{i,i-1} - f_{i+1,i} + m_i g = 0 \\ f_{i,i-1} = f_{i+1,i} + m_i g \end{aligned}$$

$$n_{i,i-1} = n_{i+1,1} + (r_i \times f_{i,i+1}) - (*r_{com(i),i} \times m_i g)$$

We iterate from i = n to 1, and we calculate the forces first.
Lung wen tsai 266-268

Chapter 4

Velocity and force ellipse

We take the force vectors and the end effector velocity, $\dot{X}$

We put the constraint that $\dot{X}^T\dot{X} = 1$

For a 2R planar, this gives us the equation of a unit circle in task space For a 3R planar, this gives us the equation of a unit sphere in task space

$$\begin{aligned} \dot{X}^T\dot{X} &= 1 \\ \dot{q}^T J^T J\dot{q} &= 1 \end{aligned}$$

The Jacobian is an expression of the configuration, the robot's shape and orientation.

A property of the Jacobian is that it is symmetric positive semi-definite, this implies the following

- The eigenvalues of $J^T J$ are non-negative
- Eigenvalues tell the length of major axis and minor axis value of the ellipse in joint space.
- Eigenvector tells the direction of the major and minor axis of the ellipse in joint space

The ellipse constantly changes as the jacobian changes.

But there are points in the robot's workspace where the ellipse becomes a perfect circle. These are known as **'isotropic points'**

And at points of singularity, the ellipse flattens.

Similarly, We can do this for

$$\tau = J^T F$$

4.1 Finding the locus of isotropic points

At isotropic points, the we find the eigenvalues of the jacobian. then we equate the eigenvalues such that

$$\frac{1}{\lambda_1} = \frac{1}{\lambda_2} = 1$$

so that the major and minor axis are equal to form a circle.

Similarly the eigenvectors define the orientation.

Chapter 5

Dynamics

It's the study of systems in motion under the influence of external forces and torques.

What forces and torques are required to achieve a given (desired) motion of a robot.

For a general second order system

$$m\ddot{x} + b\dot{x}cx = F(t)$$

This is true for a general robotic manipulator,

$$M\ddot{q} + C\dot{q} + G = \tau$$

We get this in matrix form.

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau$$

Where, M is the inertia matrix. This inertia matrix contains the moments of inertia of each joint. $C(q, \dot{q})$ is the velocity coupling vector

- Inertia matrix $M(q)$
  - tells about the mass distribution of the robotic manipulator
  - positive semi-definite
  - always iverttible
- Velocity Coupling Vector $C(q, \dot{q})$
  - Velocity squared - centripetal acceleration, q_1^2
  - cross velocity multiplication, $\dot{q}_1\dot{q}_2$
- Gravitational Force Vector $G(q)$
- tau is nothing but $\tau + J^T F$

$$D(q)\ddot{q} + H(q, \dot{q}) + G(q) = \tau$$

Part II

Practice Sheets

The notations used in our lectures are similar with this book.

- For Rotation Matrices, Pose of a rigid body, Euler Angles, Homogeneous Transformation —> read Chapter 2
- For DH Parameters and Coordinate Convention —> read Chapter 3, Sections 3.1 to 3.4

Part III

End-Sem Project

1. Select a robotic system
    (a) select only serial chain robotic manipulator
    (b) select standard SCM
    (c) No more than 5 or 6 joints
    (d) Robotic SCM can be as simple as planar 2R manipulator or complex as Stanford Manipulator
    (e) The simpler the robotic manipulator, problem should be detailed
    (f) No more than 2 teams should select the same robotic system, but they should work on different problems.
2. Problems

Problems	Type	Examples
Forward Kinematics	Compulsory	Completely define the robotic system using DH Parameters. Find homogeneous transformation and jacobian matrix Give arbitrary end-effector position by user and robotic SCM should rea
Inverse Kinematics	Compulsory	Find the workspace and show it graphically on simulation Take up some inverse kinematics problems on your will. Examples- Trajectory planning, stylus writing manpulator, dance moves performed
Statics	Optional	
Dynamics	Optional	
Controls	Optional	

3. Software
    (a) Matlab - scripts/functions, simscabe, rigidbody, multibody tool
    (b) CoppeliaSim

Part IV

Dynamics

Part V

Syllabus

Chapter 2 - Pose of a rigid body

- Position + Orientation
- Rotation Matrices
- Homogeneous Transformation Matrix

Chapter 3 - 3.1 to 3.4 - DH Parameters+Co-ordinate Convention

(Tj211.T75) Lung-Wen Tsai Segment on Direct And Inverse Kinematics John J Craig On Forward And Inverse Kinematics

- Forward Kinematics
- Inverse Kinematics

4.5-4.6.1

- Jacobian
- Singularity

Part VI

Project

Chapter 6

Robotics Project

6.1 Forward Kinematics

Find the homogeneous transformation matrices, find jacobian(pen and paper). Simulate inputting joint space variables to give the end effector's point. The simulation should show the end effector moving to that point (code)

6.2 Inverse Kinematics

Select two points within the workspace and trace the trajectory of the robot between those two points or find the workspace and plot it.

6.3 Statics

Simulate a robot under some forces and the robot maintains its equilibrium. Come up with some creative problems, such as the robot is holding a heavy object and it applies a force to hold the object against gravity.

6.4 Dynamics

1. Forward Knowing the joint torques and the external forces/torques, find the manipulator's trajectory $q(t), \dot{q(t)}, \ddot{q(t)}$
2. Inverse Find the joint torques required to move a robot in a specific trajectory. Find the trajectory of a robot with a specific joint torque vector.

6.5 Report Is Required

You can use a MATLAB live script or $\LaTeX$

Chapter 7

Modelling Project

7.1 Modelling

Make FBD and model it with analytical method.

7.2 Dynamical Response To Standard Input

Compare the robot's response to step, ramp and impulse input.

7.3 Frequency Response

Apply a sinusoidal input, and bode plot, interpret the results.