5ELEN018W - Robotic Principles
Lecture 4: Kinematics

Dr Dimitris C. Dracopoulos

More On Transformations

- Transformations of a frame (object, or point) which are relative to the fixed reference frame

More On Transformations

- Transformations of a frame (object, or point) which are relative to the fixed reference frame

Pre-multiply the transformation matrix with the coordinates described in the moving frame:

More On Transformations

- Transformations of a frame (object, or point) which are relative to the fixed reference frame

Pre-multiply the transformation matrix with the coordinates described in the moving frame:

$$\boldsymbol{p}_{xyz} = \boldsymbol{Transf}_{xyz} \times \boldsymbol{p}_{x'y'z'} \tag{1}$$

where xyz is the fixed reference frame and $x'y'z'$ is the moving frame.

More On Transformations

- Transformations of a frame (object, or point) which are relative to the fixed reference frame

Pre-multiply the transformation matrix with the coordinates described in the moving frame:

$$\boldsymbol{p}_{xyz} = \boldsymbol{Transf}_{xyz} \times \boldsymbol{p}_{x'y'z'} \tag{1}$$

where xyz is the fixed reference frame and $x'y'z'$ is the moving frame.
e.g. for a rotation about the z axis followed by a translation about the x axis, followed by a rotation about the y axis:

More On Transformations

- Transformations of a frame (object, or point) which are relative to the fixed reference frame

Pre-multiply the transformation matrix with the coordinates described in the moving frame:

$$\boldsymbol{p}_{xyz} = \boldsymbol{Transf}_{xyz} \times \boldsymbol{p}_{x'y'z'} \tag{1}$$

where xyz is the fixed reference frame and $x'y'z'$ is the moving frame.
e.g. for a rotation about the z axis followed by a translation about the x axis, followed by a rotation about the y axis:

$$\boldsymbol{Transform}_{xyz} = \boldsymbol{R}_y \cdot \boldsymbol{T}_x \cdot \boldsymbol{R}_z$$

More On Transformations (cont'd)

- Transformations of a frame (object, or point) which are relative to the moving reference frame

More On Transformations (cont'd)

- Transformations of a frame (object, or point) which are relative to the moving reference frame

Post-multiply the transformation matrix with the coordinates described in the moving (current) frame:

More On Transformations (cont'd)

- Transformations of a frame (object, or point) which are relative to the moving reference frame

Post-multiply the transformation matrix with the coordinates described in the moving (current) frame:

$$\boldsymbol{p}_{xyz} = \boldsymbol{Transf}_{x'y'z'} \times \boldsymbol{p}_{x'y'z'} \quad (2)$$

where xyz is the fixed reference frame and $x'y'z'$ is the moving frame.

More On Transformations (cont'd)

- Transformations of a frame (object, or point) which are relative to the moving reference frame

Post-multiply the transformation matrix with the coordinates described in the moving (current) frame:

$$\boldsymbol{p}_{xyz} = \boldsymbol{Transf}_{x'y'z'} \times \boldsymbol{p}_{x'y'z'} \tag{2}$$

where xyz is the fixed reference frame and $x'y'z'$ is the moving frame.
e.g. for a rotation about the z' axis followed by a translation about the x' axis, followed by a rotation about the y' axis:

More On Transformations (cont'd)

- Transformations of a frame (object, or point) which are relative to the moving reference frame

Post-multiply the transformation matrix with the coordinates described in the moving (current) frame:

$$\boldsymbol{p}_{xyz} = \boldsymbol{Transf}_{x'y'z'} \times \boldsymbol{p}_{x'y'z'} \tag{2}$$

where xyz is the fixed reference frame and $x'y'z'$ is the moving frame.
e.g. for a rotation about the z' axis followed by a translation about the x' axis, followed by a rotation about the y' axis:

$$\boldsymbol{Transform}_{x'y'z'} = \boldsymbol{R}_{z'} \cdot \boldsymbol{T}_{x'} \cdot \boldsymbol{R}_{y'}$$

Forward Kinematics vs Inverse Kinematics

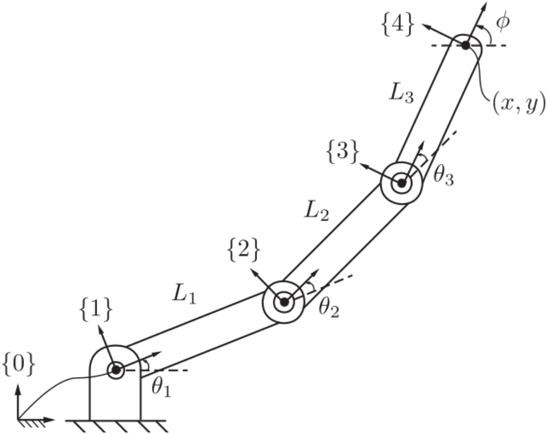
Forward Kinematics vs Inverse Kinematics

- *Forward Kinematics*: the calculation of the position and orientation of a robot's end-effector from its joint coordinates θ.

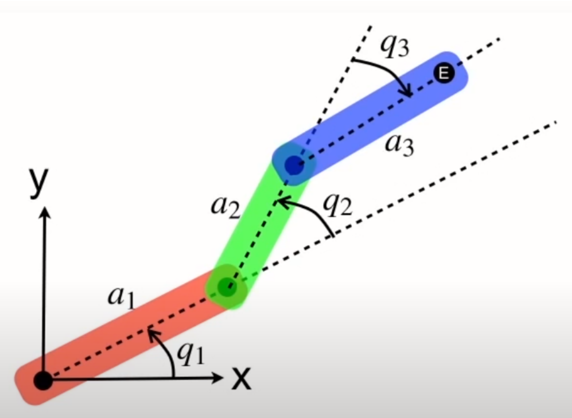
Forward Kinematics vs Inverse Kinematics

- *Forward Kinematics*: the calculation of the position and orientation of a robot's end-effector from its joint coordinates θ.
- *Inverse Kinematics*: given a position and orientation of a robot's end-effector, calculate the angles θ of the joints.

Forward Kinematics



Forward Kinematics



Representation of Configuration Space of a Robot

The position and orientation of all links.

Representation of Configuration Space of a Robot

The position and orientation of all links.
The pose of the end-effector (i.e. location and orientation) can be described with basic transformation matrices that can be multiplied together to get the homogeneous matrix.

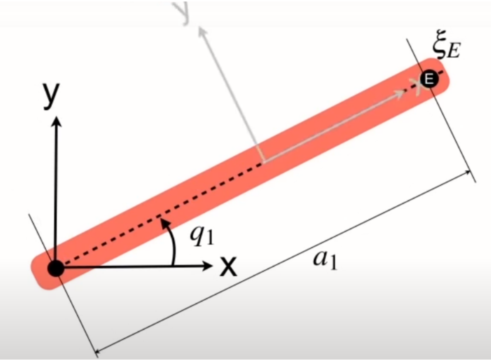
Representation of Configuration Space of a Robot

The position and orientation of all links.
The pose of the end-effector (i.e. location and orientation) can be described with basic transformation matrices that can be multiplied together to get the homogeneous matrix.

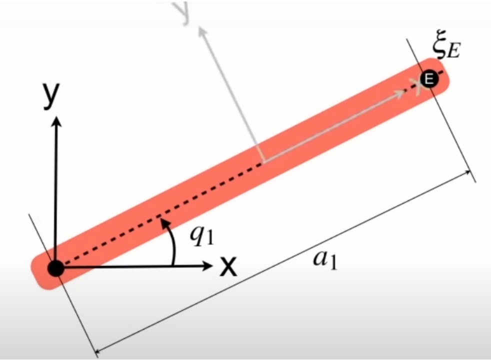
$$HomogeneousMatrix = Transf_1 * Transf2 * Transf3 * \ldots * Transf_n \quad (3)$$

where n is the number of links (assuming that each of these matrices is the <u>total</u> transformation for each link).

Example of a 1-joint Robot Arm

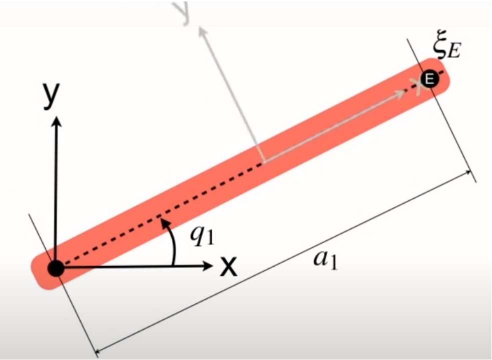


Example of a 1-joint Robot Arm



Rotation by angle q_1 and then translation by a_1.

Example of a 1-joint Robot Arm



Rotation by angle q_1 and then translation by a_1.
The homogeneous transformation describing the overall result can be calculated using the following:

$$EndEffector = Rot(q_1) \cdot T_x(a_1) \tag{4}$$

Pose of the End-Effector - 1-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

Pose of the End-Effector - 1-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1
```

Pose of the End-Effector - 1-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1

>> e = ETS2.Rz('q1')*ETS2.Tx(a1)
```

Pose of the End-Effector - 1-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

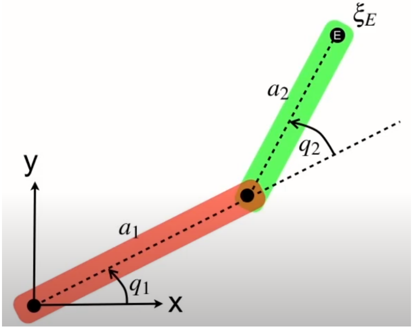
```
>> a1 = 1

>> e = ETS2.Rz('q1')*ETS2.Tx(a1)

>> e.fkine(pi/2)
```

Pose of the End-Effector - 1-Joint 2D Robot Arm (cont'd)

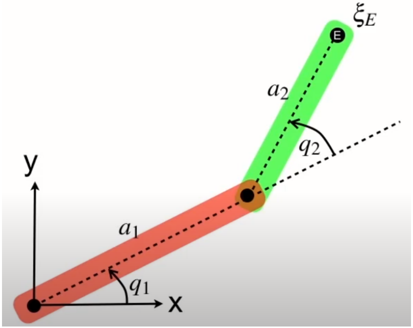
In Matlab Robotics Toolbox:

```
>> a1 = 1

>> e = ETS2.Rz('q1')*ETS2.Tx(a1)

>> e.fkine(pi/2)

>> e.teach
```

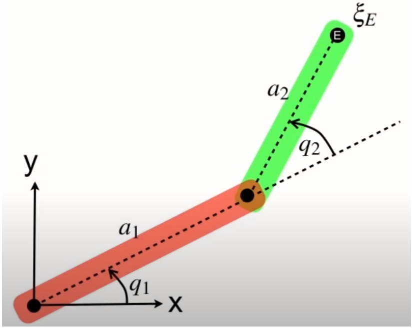
Example of a 2-joint Planar Robot Arm



Example of a 2-joint Planar Robot Arm

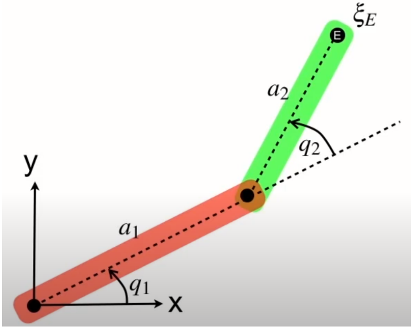


Example of a 2-joint Planar Robot Arm



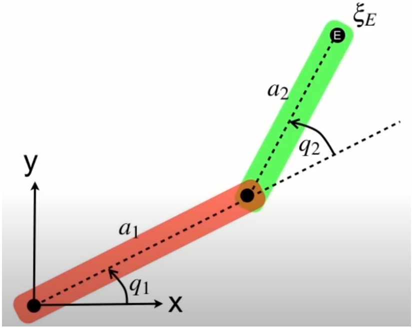
1. Rotation by angle q_1

Example of a 2-joint Planar Robot Arm



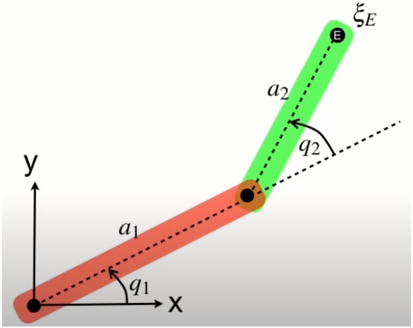
1. Rotation by angle q_1
2. Translation by a_1

Example of a 2-joint Planar Robot Arm



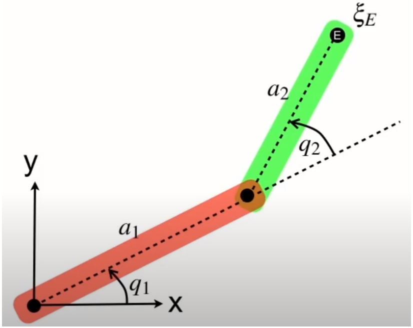
1. Rotation by angle q_1
2. Translation by a_1
3. Rotation by angle q_2

Example of a 2-joint Planar Robot Arm



1. Rotation by angle q_1
2. Translation by a_1
3. Rotation by angle q_2
4. Translation by a_2

Example of a 2-joint Planar Robot Arm



1. Rotation by angle q_1
2. Translation by a_1
3. Rotation by angle q_2
4. Translation by a_2

The homogeneous transformation describing the overall result can be calculated using the following:

$$EndEffector = Rot(q_1) \cdot T_x(a_1) \cdot Rot(q_2) \cdot T_x(a_2) \quad (5)$$

Pose of the End-Effector - 2-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

Pose of the End-Effector - 2-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1;
```

Pose of the End-Effector - 2-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1;
>> a2 = 1;
```

Pose of the End-Effector - 2-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1;
>> a2 = 1;

>> e = ETS2.Rz('q1')*ETS2.Tx(1)*ETS2.Rz('q2')*ETS2.Tx(a2)
```

Pose of the End-Effector - 2-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1;
>> a2 = 1;

>> e = ETS2.Rz('q1')*ETS2.Tx(1)*ETS2.Rz('q2')*ETS2.Tx(a2)

>> e.fkine([pi/2 pi])
```

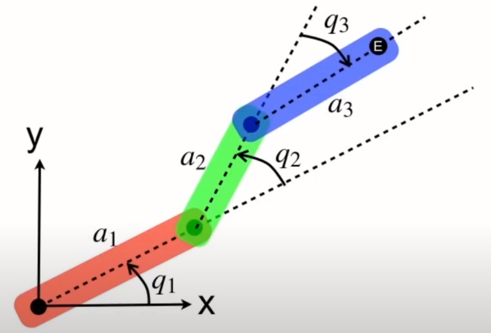
Pose of the End-Effector - 2-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

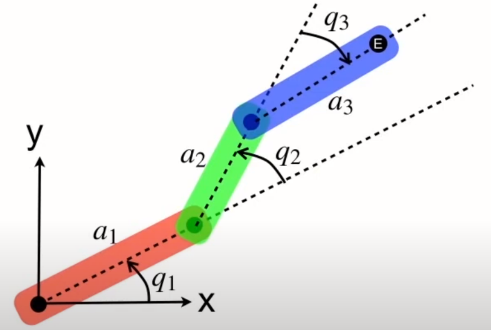
```
>> a1 = 1;
>> a2 = 1;

>> e = ETS2.Rz('q1')*ETS2.Tx(1)*ETS2.Rz('q2')*ETS2.Tx(a2)

>> e.fkine([pi/2 pi])

>> e.teach
```

Example of a 3-joint Planar Robot Arm

Example of a 3-joint Planar Robot Arm



Example of a 3-joint Planar Robot Arm



The homogeneous transformation describing the overall result can be calculated using the following:

$$EndEffector = Rot(q_1) \cdot T_x(a_1) \cdot Rot(q_2) \cdot T_x(a_2) \cdot Rot(q_3) \cdot T_x(a_3)$$

Pose of the End-Effector - 3-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

Pose of the End-Effector - 3-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1
```

Pose of the End-Effector - 3-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1
>> a2 = 1
```

Pose of the End-Effector - 3-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1
>> a2 = 1
>> a3 = 1
```

Pose of the End-Effector - 3-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1
>> a2 = 1
>> a3 = 1

>> e = ETS2.Rz('q1')*ETS2.Tx(a1)*ETS2.Rz('q2')*ETS2.Tx(a2) ...
          *ETS2.Rz('q3')*ETS2.Tx(a3)
```

Pose of the End-Effector - 3-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1
>> a2 = 1
>> a3 = 1

>> e = ETS2.Rz('q1')*ETS2.Tx(a1)*ETS2.Rz('q2')*ETS2.Tx(a2) ...
          *ETS2.Rz('q3')*ETS2.Tx(a3)

>> e.fkine([pi pi/2 pi/4])
```

Pose of the End-Effector - 3-Joint 2D Robot Arm (cont'd)

In Matlab Robotics Toolbox:

```
>> a1 = 1
>> a2 = 1
>> a3 = 1

>> e = ETS2.Rz('q1')*ETS2.Tx(a1)*ETS2.Rz('q2')*ETS2.Tx(a2) ...
          *ETS2.Rz('q3')*ETS2.Tx(a3)

>> e.fkine([pi pi/2 pi/4])

>> e.teach
```

The Problem of Forward Kinematics

The calculation of the position and orientation of a robot's end-effector from its joint coordinates θ_i.

The Problem of Forward Kinematics

The calculation of the position and orientation of a robot's end-effector from its joint coordinates θ_i.

- In the previous slides it has been shown how to do this in 2D spaces for:

The Problem of Forward Kinematics

The calculation of the position and orientation of a robot's end-effector from its joint coordinates θ_i.

- In the previous slides it has been shown how to do this in 2D spaces for:
  - $\rightarrow$ 1-joint robot arms

The Problem of Forward Kinematics

The calculation of the position and orientation of a robot's end-effector from its joint coordinates θ_i.

- In the previous slides it has been shown how to do this in 2D spaces for:
  - $\rightarrow$ 1-joint robot arms
  - $\rightarrow$ 2-joint robot arms

The Problem of Forward Kinematics

The calculation of the position and orientation of a robot's end-effector from its joint coordinates θ_i.

- In the previous slides it has been shown how to do this in 2D spaces for:
  - $\rightarrow$ 1-joint robot arms
  - $\rightarrow$ 2-joint robot arms
  - $\rightarrow$ 3-joint robot arms

using simple transformations in Mathematics which correspond to real operations in Physics!

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

- Reduces the relationship between 2 coordinate frames from 6 parameters to 4 parameters.

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

- Reduces the relationship between 2 coordinate frames from 6 parameters to 4 parameters.
- Each joint in a robot is described by 4 parameters.

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

- Reduces the relationship between 2 coordinate frames from 6 parameters to 4 parameters.
- Each joint in a robot is described by 4 parameters.

How is this achieved?

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

- Reduces the relationship between 2 coordinate frames from 6 parameters to 4 parameters.
- Each joint in a robot is described by 4 parameters.

How is this achieved?
The coordinate frames have constraints.

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

- Reduces the relationship between 2 coordinate frames from 6 parameters to 4 parameters.
- Each joint in a robot is described by 4 parameters.

How is this achieved?

The coordinate frames have constraints.

- x axis of frame j intersects the z axis of frame $j-1$

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

- Reduces the relationship between 2 coordinate frames from 6 parameters to 4 parameters.
- Each joint in a robot is described by 4 parameters.

How is this achieved?

The coordinate frames have constraints.

- x axis of frame j intersects the z axis of frame $j-1$
- x axis of frame j is perpendicular to the z axis of frame $j-1$

The Denavit-Hartenberg (DH) Notation

The relationship between two coordinate frames is described by 6 parameters (3 translations and 3 rotations). Can this be improved?

Attach a coordinate frame to the end of each link.

- Reduces the relationship between 2 coordinate frames from 6 parameters to 4 parameters.
- Each joint in a robot is described by 4 parameters.

How is this achieved?

The coordinate frames have constraints.

- x axis of frame j intersects the z axis of frame $j-1$
- x axis of frame j is perpendicular to the z axis of frame $j-1$

$\longrightarrow$ 6 parameters - 2 constraints means 4 parameters are needed.

The Denavit-Hartenberg (DH) Notation (cont'd)

4 parameters used associated with each link i and joint i:

The Denavit-Hartenberg (DH) Notation (cont'd)

4 parameters used associated with each link i and joint i:

- $\boldsymbol{\theta_i}$: joint angle

The Denavit-Hartenberg (DH) Notation (cont'd)

4 parameters used associated with each link i and joint i:

- $\boldsymbol{\theta_i}$: joint angle
- $\boldsymbol{d_i}$: link offset

The Denavit-Hartenberg (DH) Notation (cont'd)

4 parameters used associated with each link i and joint i:

- $\boldsymbol{\theta_i}$: joint angle
- $\boldsymbol{d_i}$: link offset
- $\boldsymbol{r_i}$ (or $\boldsymbol{a_i}$ in most textbooks): link length

The Denavit-Hartenberg (DH) Notation (cont'd)

4 parameters used associated with each link i and joint i:

- $\boldsymbol{\theta_i}$: joint angle
- $\boldsymbol{d_i}$: link offset
- $\boldsymbol{r_i}$ (or $\boldsymbol{a_i}$ in most textbooks): link length
- $\boldsymbol{\alpha_i}$: link twist

The Denavit-Hartenberg (DH) Notation (cont'd)

4 parameters used associated with each link i and joint i:

- $\boldsymbol{\theta_i}$: joint angle
- $\boldsymbol{d_i}$: link offset
- $\boldsymbol{r_i}$ (or $\boldsymbol{a_i}$ in most textbooks): link length
- $\boldsymbol{\alpha_i}$: link twist

The Denavit-Hartenberg (DH) Notation (cont'd)

4 parameters used associated with each link i and joint i:

- $\boldsymbol{\theta_i}$: joint angle
- $\boldsymbol{d_i}$: link offset
- $\boldsymbol{r_i}$ (or $\boldsymbol{a_i}$ in most textbooks): link length
- $\boldsymbol{\alpha_i}$: link twist

Each homogeneous transformation A_i is represented as the product of 4 basic transformations:

$$A_i = Rot_{z,\theta_i} \cdot Trans_{z,d_i} \cdot Trans_{x,r_i} \cdot Rot_{x,\alpha_i} \tag{6}$$

The Denavit-Hartenberg (DH) Notation (cont'd)

$$A_i = Rot_{z,\theta_i} \cdot Trans_{z,d_i} \cdot Trans_{x,r_i} \cdot Rot_{x,\alpha_i} =$$

The Denavit-Hartenberg (DH) Notation (cont'd)

$$A_i = Rot_{z,\theta_i} \cdot Trans_{z,d_i} \cdot Trans_{x,r_i} \cdot Rot_{x,\alpha_i} =$$

$$\begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\times \begin{bmatrix} 1 & 0 & 0 & r_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

The Denavit-Hartenberg (DH) Notation (cont'd)

$$A_i = Rot_{z,\theta_i} \cdot Trans_{z,d_i} \cdot Trans_{x,r_i} \cdot Rot_{x,\alpha_i} =$$

$$\begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\times \begin{bmatrix} 1 & 0 & 0 & r_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & r_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & r_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

The DH Table

The DH notation requires a table. The number of rows equals the number of joints and it has 4 columns each one corresponding to the 4 parameters for the joint i of that row.

The DH Table

The DH notation requires a table. The number of rows equals the number of joints and it has 4 columns each one corresponding to the 4 parameters for the joint i of that row.

For example for a robot with 4 joints:

The DH Table

The DH notation requires a table. The number of rows equals the number of joints and it has 4 columns each one corresponding to the 4 parameters for the joint i of that row.

For example for a robot with 4 joints:

Joint	$\boldsymbol{\theta}$	$\boldsymbol{r}$	$\boldsymbol{d}$	$\boldsymbol{\alpha}$
1	θ_1	r_1	d_1	α_1
2	θ_2	r_2	d_2	α_2
3	θ_3	r_3	d_3	α_3
4	θ_4	r_4	d_4	α_4

The DH Table

The DH notation requires a table. The number of rows equals the number of joints and it has 4 columns each one corresponding to the 4 parameters for the joint i of that row.

For example for a robot with 4 joints:

Joint	$\boldsymbol{\theta}$	$\boldsymbol{r}$	$\boldsymbol{d}$	$\boldsymbol{\alpha}$
1	θ_1	r_1	d_1	α_1
2	θ_2	r_2	d_2	α_2
3	θ_3	r_3	d_3	α_3
4	θ_4	r_4	d_4	α_4

- For *revolute joints*: Only θ changes, all the other 3 parameters are fixed according to the robot mechanism.

The DH Table

The DH notation requires a table. The number of rows equals the number of joints and it has 4 columns each one corresponding to the 4 parameters for the joint i of that row.

For example for a robot with 4 joints:

Joint	$\boldsymbol{\theta}$	$\boldsymbol{r}$	$\boldsymbol{d}$	$\boldsymbol{\alpha}$
1	θ_1	r_1	d_1	α_1
2	θ_2	r_2	d_2	α_2
3	θ_3	r_3	d_3	α_3
4	θ_4	r_4	d_4	α_4

- For *revolute joints*: Only θ changes, all the other 3 parameters are fixed according to the robot mechanism.
- For *prismatic joints*: Only d changes, all the other 3 parameters are fixed according to the robot mechanism.

The DH Table

The DH notation requires a table. The number of rows equals the number of joints and it has 4 columns each one corresponding to the 4 parameters for the joint i of that row.

For example for a robot with 4 joints:

Joint	$\boldsymbol{\theta}$	$\boldsymbol{r}$	$\boldsymbol{d}$	$\boldsymbol{\alpha}$
1	θ_1	r_1	d_1	α_1
2	θ_2	r_2	d_2	α_2
3	θ_3	r_3	d_3	α_3
4	θ_4	r_4	d_4	α_4

- For *revolute joints*: Only θ changes, all the other 3 parameters are fixed according to the robot mechanism.
- For *prismatic joints*: Only d changes, all the other 3 parameters are fixed according to the robot mechanism.

Example of DH Notation

Consider the following DH table:

Joint	$\boldsymbol{\theta}$	$\boldsymbol{r}$	$\boldsymbol{d}$	$\boldsymbol{\alpha}$
1	π	5	2	$\frac{\pi}{2}$

Example of DH Notation

Consider the following DH table:

Joint	$\boldsymbol{\theta}$	$\boldsymbol{r}$	$\boldsymbol{d}$	$\boldsymbol{\alpha}$
1	π	5	2	$\frac{\pi}{2}$

What is the DH matrix which corresponds to the above table?

Example of DH Notation

Consider the following DH table:

Joint	$\boldsymbol{\theta}$	$\boldsymbol{r}$	$\boldsymbol{d}$	$\boldsymbol{\alpha}$
1	π	5	2	$\frac{\pi}{2}$

What is the DH matrix which corresponds to the above table?

Answer:

```
  -1.0000   -0.0000   -0.0000   -5.0000
   0.0000   -0.0000    1.0000    0.0000
        0    1.0000    0.0000    2.0000
        0         0         0    1.0000
```

Example of DH Notation

Consider the following DH table:

Joint	θ	r	d	α
1	π	5	2	$\frac{\pi}{2}$

What is the DH matrix which corresponds to the above table?

Answer:

```
   -1.0000   -0.0000   -0.0000   -5.0000
    0.0000   -0.0000    1.0000    0.0000
         0    1.0000    0.0000    2.0000
         0         0         0    1.0000
```

To calculate, apply Equation (7).

Finding the Pose of the End-Effector relative to the Base Frame

Assume that $A_1, A_2, A_3 \ldots A_n$ are the DH matrices of all the robot joints $1, 2, 3, \ldots n$.

Finding the Pose of the End-Effector relative to the Base Frame

Assume that $A_1, A_2, A_3 \ldots A_n$ are the DH matrices of all the robot joints $1, 2, 3, \ldots n$.
Then the calculation requires the multiplication of all the matrices:

$$Pose_{end\_effector} = A_1 \cdot A_2 \cdot A_3 \ldots A_n \qquad (8)$$

Example: Calculation of the Pose of the End-Effector

The following DH matrices correspond to the joints of a robot, from robot base to end-effector. Find the pose of the end-effector relative to the robot base.

$$A_1 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} -1 & 0 & 0 & -2 \\ 0 & -0 & 1 & 0 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Example: Calculation of the Pose of the End-Effector (cont'd)

Example: Calculation of the Pose of the End-Effector (cont'd)

Simply calculate $A_1 * A_2 * A_3$.