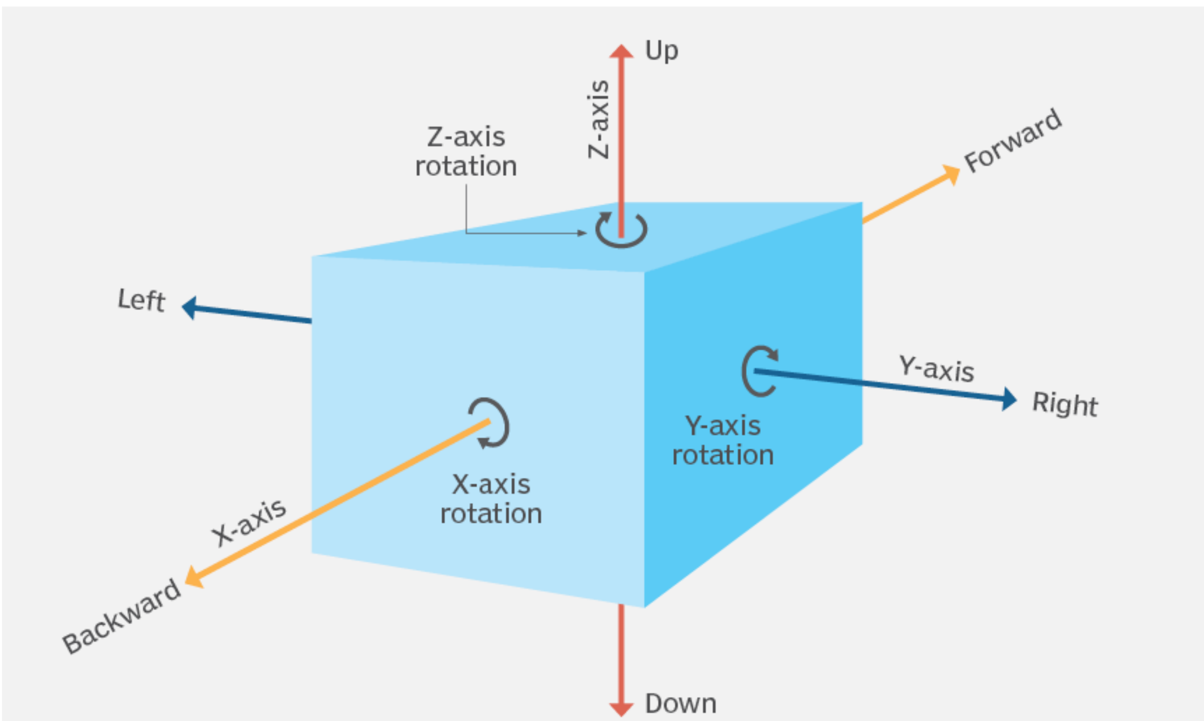
Fundamentals of robotics

Iheb Bouariche | Robotics | 24-12-2022

Degree of freedom:

Degrees of freedom refers to the maximum number of logically independent values, which are values that have the freedom to vary, in the data sample.

In mechanics, degrees of freedom (DOF) are the number of independent variables that define the possible positions or motions of a mechanical system in space. DOF measurements assume that the mechanism is both rigid and unconstrained, whether it operates in two-dimensional or three-dimensional space. Degrees of freedom applies to two types of motion: translational and rotational. An object in three-dimensional space can support up to six degrees of freedom: three translational and three rotational. Because it is in a three-dimensional space, the block can move linearly along all three axes, and it can rotate around all three axes. (in two-dimensional space, an object can support 3 degrees of freedom: two translational and one rotational)



Representation of an Object in 3D space:

An object in space can be represented in position (3 coordinates) and orientation (3 coordinates).

1. **Position representation:** it's by representing the position of the mass center of an object on the 3 axes of a frame "n" (x, y, z) by a [3x1] vector. It's named also by Translation vector.
2. **Orientation representation:** it's by representing the orientation of an object or an axis in the 3D space by using [3x3] matrix and it refers to the imaginary rotation that is needed to move the object from a reference placement to its current placement.

The rotation matrix R can be defined as a representation of the body frame unit axes expressed in the base frame as:

$$R = [\,\hat{x}_b \quad \hat{y}_b \quad \hat{z}_b\,] = \begin{pmatrix} \hat{x}_b.\,\hat{x}_s & \hat{y}_b.\,\hat{x}_s & \hat{z}_b.\,\hat{x}_s \\ \hat{x}_b.\,\hat{y}_s & \hat{y}_b.\,\hat{y}_s & \hat{z}_b.\,\hat{y}_s \\ \hat{x}_b.\,\hat{z}_s & \hat{y}_b.\,\hat{z}_s & \hat{z}_b.\,\hat{z}_s \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

Where xb.xs is the representation of xs axis of the oriented body in the xb axis of the base frame. and we do the same thing for the rest...

Notes:

- In the 3D representation of an object or an axis, we prefer to start by the orientation to the current rotations (there are multiple representations), then, translation of the object to the current position.
- An object in the 3D space defined by an axis.
- We know that we have only 3 variables for orientation (3-DOF on orientation), and that means, the nine numbers in this rotation matrix must have six constraints. these constraints are provided by the orthogonality condition of the object axis , the constraints are proved by the dot product of every column vector of the rotation matrix with another is zero.

$$\begin{cases} r_{11}r_{12} + r_{21}r_{22} + r_{31}r_{32} = 0 \\ r_{11}r_{13} + r_{21}r_{23} + r_{31}r_{33} = 0 \\ r_{12}r_{13} + r_{22}r_{23} + r_{32}r_{33} = 0 \end{cases}$$

Homogeneous transformation matrix frame:

A frame is a set of four vectors carrying positions and orientation information.

Homogeneous matrix is a frame that package the rotation matrix and the translation vector in one [4x4] matrix.

$$T = \begin{pmatrix} R & p \\ o & 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & p_1 \\ r_{21} & r_{22} & r_{23} & p_2 \\ r_{31} & r_{32} & r_{33} & p_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

The inverse of a transformation matrix T is also a transformation matrix and can be computed as:

$$T^{-1} = \begin{pmatrix} R & p \\ O & 1 \end{pmatrix}^{-1} = \begin{pmatrix} R^T & -R^T p \\ O & 1 \end{pmatrix} \in SE(3)$$

We can use the homogeneous transformation on:

- T is used for transforming the point x to Tx, this means that the point x is rotated by R and translated by p as Rx + p, therefore Tx means the representation of x in homogenous coordinates as:

$$T\begin{pmatrix} x \\ 1 \end{pmatrix} = \begin{pmatrix} R & p \\ o & 1 \end{pmatrix}\begin{pmatrix} x \\ 1 \end{pmatrix} = \begin{pmatrix} Rx + p \\ 1 \end{pmatrix}$$

- The homogeneous transformation can be applied to transform (orientate and translate) every object or axes in the space, also it can provide a representation of points, vectors, objects from an axis to another.
- We can multiply two or more than two homogeneous matrices (applied in the case of rotation matrix) to find another end-to-end transformation matrix. For example:

T(i-1)(i) x T(i)(i+1) = T(i-1)(i+2).

Translation operator:

Trans $(\hat{X}, q)$: Translation of q units along x-direction

$$\text{Trans}\,(\hat{X}, q) = \begin{bmatrix} 1 & 0 & 0 & q \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Note: Trans operators are commutative in nature

$$Trans(\hat{X}, q_x)Trans(\hat{Y}, q_y) = Trans(\hat{Y}, q_y)Trans(\hat{X}, q_x)$$

Rotational operator:

$$\begin{bmatrix} q_X \\ q_Y \\ q_Z \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix}$$

$$\mathrm{Rot}(\hat{Z},\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Similarly, we get

$$\mathrm{Rot}(\hat{X},\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

$$\mathrm{Rot}(\hat{Y},\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

- Rotation matrices are not commutative in nature

$$ROT(\hat{X},\theta_1)ROT(\hat{Y},\theta_2) \neq ROT(\hat{Y},\theta_2)ROT(\hat{X},\theta_1)$$

- Inverse of a rotation matrix is nothing but its transpose

$$ROT^{-1}(\hat{X},\theta) = ROT^{T}(\hat{X},\theta)$$

- ${}^{A}_{B}T = {}^{B}_{A}T^{-1}$

- Representation of position O can be represented by Cartesian representation, cylindrical represented or spherical representation.
- Representation of matrix R (position of rotated axe) can be represented by cartesian representation, Roll, Pitch and Yaw representation, Euler representation.

Notes: that all these representations are used to solve cartesian position by other representation, and also for cartesian rotation by angles representation.

For example: - I have a homogeneous matrix that contain the cartesian position O, I can solve the cylindrical representation by finding the values of theta, r and z that gives us the solution O.

- Another example for orientation matrix that I have all its values, and I can solve the Euler representation by finding the angles by Z, Y and X. for example:

$${}^{B}_{U}R_{Eulerangles} = ROT(\hat{X}_{B''},-\gamma)ROT(\hat{Y}_{B'},-\beta)ROT(\hat{Z}_{B},-\alpha)$$

$${}^{U}_{B}R = \begin{bmatrix} c\alpha c\beta & s\beta s\gamma c\alpha - s\alpha c\gamma & s\beta c\gamma c\alpha + s\alpha s\gamma \\ s\alpha c\beta & s\beta s\gamma s\alpha + c\alpha c\gamma & s\beta c\gamma s\alpha - s\gamma c\alpha \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix}$$

We compare with

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

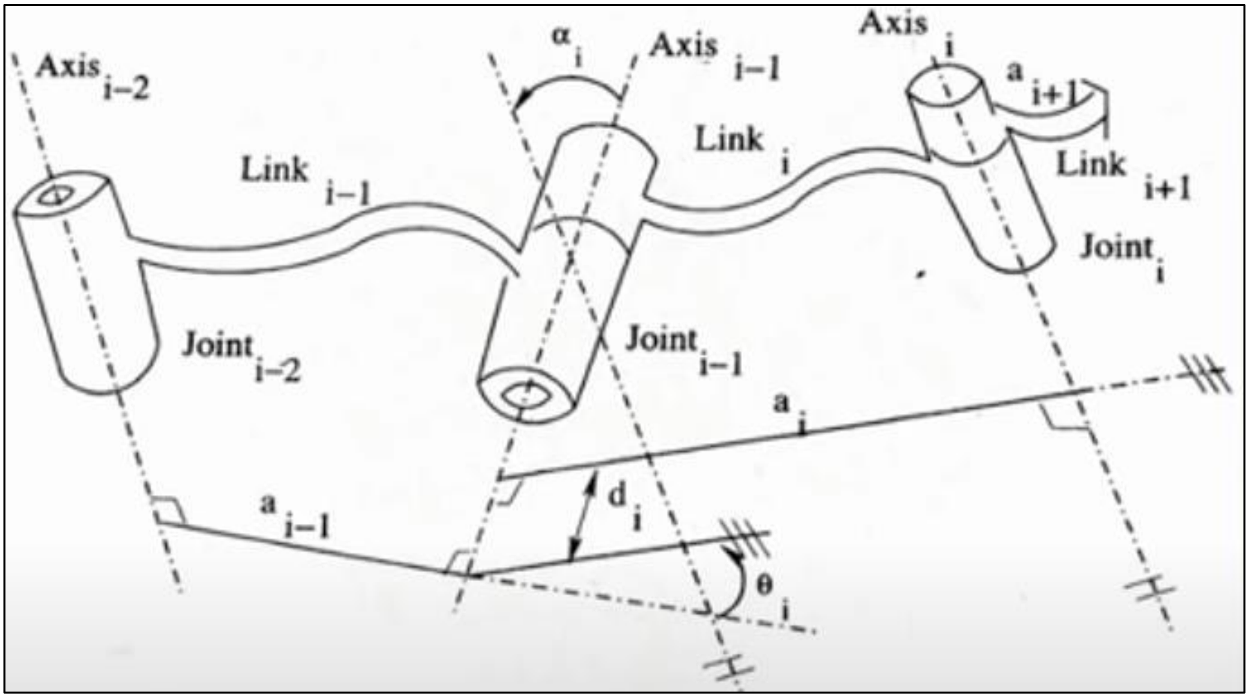
$$\alpha = \tan^{-1}\left(\frac{r_{21}}{r_{11}}\right)$$

$$\beta = \tan^{-1}\left(\frac{-r_{31}}{\sqrt{r_{11}^2 + r_{21}^2}}\right)$$

$$\gamma = \tan^{-1}\left(\frac{r_{32}}{r_{33}}\right)$$

Denavit-Hartenberg Notations: it's based on the rule that any joint system (revolute or prismatic) in a robot can be presented with four parameters (Y translation or rotation can be ignored).

DH-Notations:



1. **Length of link (a(i)):** it's the mutual perpendicular Axis(i-1) and Axis(i).
2. **Angle of twist of link (alpha(i)):** it is defined as the angle between Axis(i-1) and Axis(i).
3. **Offset of link (d(i)):** it is the distance measured from a point where a(i-1) intersects the Axis(i-1) to the point where a(i) intersect the Axis(i-1) measured along the said axis.
4. **Joint angle (theta(i)):** it is defined as the angle between the extension of a(i-1) an a(i) measured about the Axis(i-1).

DH-Rules:

1. Z(i) is an axis about which the rotation is considered or along which the translation takes places.
2. If Z(i-1) and Z(i) axes are parallel to each other, X axis will be directed from Z(i-1) to Z(i) along their common normal.
3. If Z(i-1) and Z(i) axes intersect each other, X axis can be selected along either of two remaining directions.
4. If Z(i-1) and Z(i) axes act along a straight line, X axis can be selected anywhere in a plane perpendicular to them.
5. Y axis is decided as Y = Z*X. (right-hand rule)

- By using these notations and rules, we can now find the DH-Table's values then calculate the homogeneous transformation matrix of a robot manipulator with this matrix. And then multiply all the "T" s.

$$ {}^{i-1}_{\ i}T = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Notes: - The DH-Homogeneous matrix can be calculated by:

$$
\begin{aligned}
{}^{i-1}_{i}T &= {}^{i-1}_{A}T\,{}^{A}_{B}T\,{}^{B}_{C}T\,{}^{C}_{i}T \\
&= ROT(Z,\theta_i)TRANS(Z,d_i)ROT(X,\alpha_i)TRANS(X,a_i) \\
&= Screw_Z Screw_X
\end{aligned}
$$

- by inverting the DH-Matrix we found:

Now, ${}^{i}_{i-1}T = \left[{}^{i-1}_{i}T\right]^{-1}$

$$
= \begin{bmatrix} c\theta_i & s\theta_i & 0 & -a_i \\ -s\theta_i c\alpha_i & c\theta_i c\alpha_i & s\alpha_i & -d_i s\alpha_i \\ s\theta_i s\alpha_i & -c\theta_i s\alpha_i & c\alpha_i & -d_i c\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Forward Kinematics: it can be founded by calculating the end-to-end homogeneous matrix T, so if I had the articulations values and the geometrical parameters of the system, I can find the position and the orientation of the end effector.

Inverse Kinematics: it can be founded by an analytical calculation. In this case we had the positions values and we need to found the articulations positions.

The second method is by using the orientation and position values to find the inverse kinematic (All T values are known and this method is used when we have a big number of n-DOF).

Example: We assume that we calculated the direct kinematic (end-to-end homogeneous matrix T).

$$
{}^{0}_{5}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & q_x \\ r_{21} & r_{22} & r_{23} & q_y \\ r_{31} & r_{32} & r_{33} & q_z \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
\begin{aligned}
{}^{0}_{5}T &= {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T \\
\Rightarrow {}^{0}_{1}T^{-1}({}^{0}_{5}T) &= {}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T
\end{aligned}
$$

$$\Rightarrow \begin{bmatrix} c_1 & s_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & q_x \\ r_{21} & r_{22} & r_{23} & q_y \\ r_{31} & r_{32} & r_{33} & q_z \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} c_{234}c_5 & -c_{234}s_5 & s_{234} & L_1c_2 + L_2c_{23} \\ s_{234}c_5 & -s_{234}s_5 & -c_{234} & L_1s_2 + L_2s_{23} \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} r_{11}c_1 + r_{21}s_1 & r_{12}c_1 + r_{22}s_1 & r_{13}c_1 + r_{23}s_1 & q_xc_1 + q_ys_1 \\ -r_{31} & -r_{32} & -r_{33} & -q_z \\ -r_{11}s_1 + r_{21}c_1 & -r_{12}s_1 + r_{22}c_1 & -r_{13}s_1 + r_{23}c_1 & -q_xs_1 + q_yc_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} c_{234}c_5 & -c_{234}s_5 & s_{234} & L_1c_2 + L_2c_{23} \\ s_{234}c_5 & -s_{234}s_5 & -c_{234} & L_1s_2 + L_2s_{23} \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$q_xc_1 + q_ys_1 = L_1c_2 + L_2c_{23}$$

$$-q_xs_1 + q_yc_1 = 0$$

$$q_z = -L_1s_2 - L_2s_{23}$$

$$s_{234} = r_{13}c_1 + r_{23}s_1$$

$$c_{234} = r_{33}$$

$$-r_{11}s_1 + r_{21}c_1 = s_5$$

$$-r_{12}s_1 + r_{22}c_1 = c_5$$

$$\begin{aligned} -q_xs_1 + q_yc_1 &= 0 \\ \Rightarrow \theta_1 &= \arctan(\frac{q_y}{q_x}) \end{aligned}$$

$$
\begin{aligned}
q_x^2 + q_y^2 + q_z^2 &= L_1^2 + L_2^2 + 2L_1L_2c_3 \\
\Rightarrow \theta_3 &= \arccos\left(\frac{q_x^2 + q_y^2 + q_z^2 - L_1^2 - L_2^2}{2L_1L_2}\right)
\end{aligned}
$$

$$
\begin{aligned}
L_1c_2 + L_2c_{23} &= q_xc_1 + q_ys_1 \\
\Rightarrow (L_1 + L_2c_3)c_2 - (L_2s_3)s_2 &= q_xc_1 + q_ys_1
\end{aligned}
$$

Let us assume $L_1 + L_2c_3 = \rho\sin\alpha$ and $L_2s_3 = \rho\cos\alpha$, where $\rho \neq 0$ and $\rho = \sqrt{(L_1 + L_2c_3)^2 + (L_2s_3)^2}$; $\alpha = \arctan(\frac{L_1+L_2c_3}{L_2s_3})$. Thus, the above expression can be written as follows:

$$
\rho\sin\alpha c_2 - \rho\cos\alpha s_2 = q_xc_1 + q_ys_1
$$

$$
\rho\sin(\alpha - \theta_2) = q_xc_1 + q_ys_1
$$

$$
\rho\cos(\alpha - \theta_2) = -q_z
$$

$$
\begin{aligned}
\tan(\alpha - \theta_2) &= \frac{q_xc_1 + q_ys_1}{-q_z} \\
\Rightarrow \theta_2 &= \alpha - \arctan\left(\frac{q_xc_1 + q_ys_1}{-q_z}\right)
\end{aligned}
$$

$$
\begin{aligned}
\theta_2 + \theta_3 + \theta_4 &= \arctan\left(\frac{r_{13}c_1 + r_{23}s_1}{r_{33}}\right) \\
\Rightarrow \theta_4 &= \arctan\left(\frac{r_{13}c_1 + r_{23}s_1}{r_{33}}\right) - \theta_2 - \theta_3
\end{aligned}
$$

$$
\theta_5 = \arctan\left(\frac{-r_{11}s_1 + r_{21}c_1}{-r_{12}s_1 + r_{22}c_1}\right)
$$

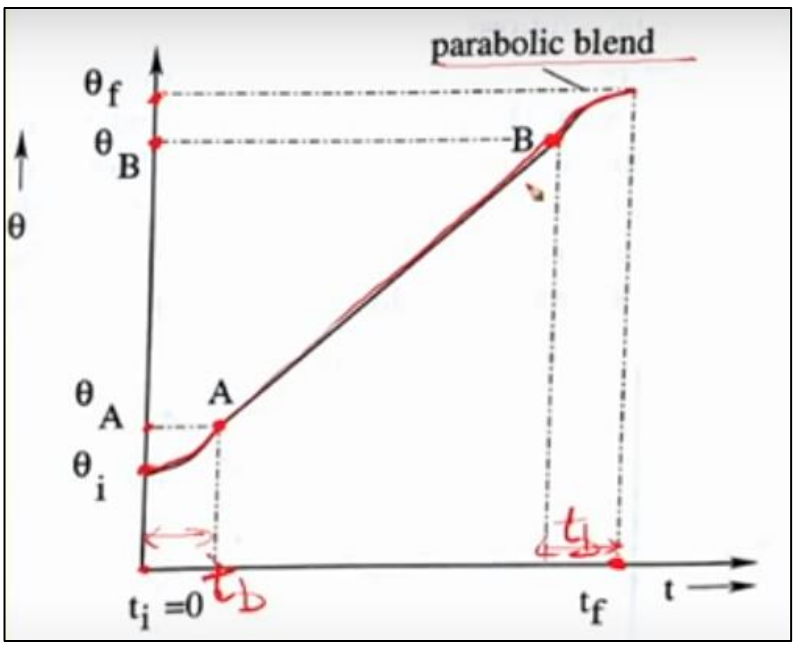
Trajectory planning: there are three trajectory functions that are used to guaranteed: the position and the derivative of the position (velocity) and the second derivative of the position (acceleration) are not equals to zero, and providing a smooth transition from a position to other. The three trajectory functions are:

- Cubic polynomial.
- Fifth-order polynomial.
- Linear trajectory function.

Note: that in practical way as a user I enter a cartesian trajectory, then the inverse kinematic model gives us the articulation position.

Note2: the current articulation of a robot and its velocity and acceleration are named in trajectory planning as the initial "q"s, and the desired position, velocity and acceleration are the finals "q"s.

1. Cubic polynomial: it's a polynomial equation, with conditions on position and velocity (in this case we use a 3-deg polynomial equation).
2. Fifth-order polynomial is used when we have conditions on position, velocity and acceleration, so we use a 5-deg polynomial equation.
3. Linear trajectory: a linear trajectory function with a parabolic blend. (check the course)



Jacobian: can be calculated by dividing the forward kinematic equation by the time.

*V(linear and angular) = J(linear and angular) * q_dot   where q_dot is the articulation velocity.*

Note: that the inverse of the Jacobian can be used for singularity checking. (det(J) =!= 0)

Question: pourquoi on detecte les points de singularités par le modele cinematique seulement? Est ce que on peut les detecter par le modele geometrique?

Inverse Dynamics:

Inertia tensor of i-th link:

$$J_i = \int {}^i_i\bar{r}\,{}^i_i\bar{r}^{T'}\, dm$$

$$= \begin{bmatrix} \int x_i^2 dm & \int x_i y_i dm & \int x_i z_i dm & \int x_i dm \\ \int x_i y_i dm & \int y_i^2 dm & \int y_i z_i dm & \int y_i dm \\ \int x_i z_i dm & \int y_i z_i dm & \int z_i^2 dm & \int z_i dm \\ \int x_i dm & \int y_i dm & \int z_i dm & \int dm \end{bmatrix}$$

$$\int dm = m$$

Inertia tensor, J_i can be written as

$$J_i = \begin{bmatrix} \frac{-I_{XX}+I_{YY}+I_{ZZ}}{2} & I_{XY} & I_{ZX} & m_i \bar{x}_i \\ I_{XY} & \frac{I_{XX}-I_{YY}+I_{ZZ}}{2} & I_{YZ} & m_i \bar{y}_i \\ I_{ZX} & I_{YZ} & \frac{I_{XX}+I_{YY}-I_{ZZ}}{2} & m_i \bar{z}_i \\ m_i \bar{x}_i & m_i \bar{y}_i & m_i \bar{z}_i & m_i \end{bmatrix}$$

Moment of inertia:

$$[I] = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}$$

$$I_{xx} = \int (y^2 + z^2)\, dm = mk_x^2$$
$$I_{yy} = \int (x^2 + z^2)\, dm = mk_y^2$$
$$I_{zz} = \int (x^2 + y^2)\, dm = mk_z^2$$
$$I_{xy} = \int xy\, dm$$
$$I_{xz} = \int xz\, dm$$
$$I_{yz} = \int yz\, dm$$

Lagrange-Euler formulation:

- Lagrange-Euler Formulation

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \tau_i$$

  – Lagrange function is defined

$$L = K - P$$

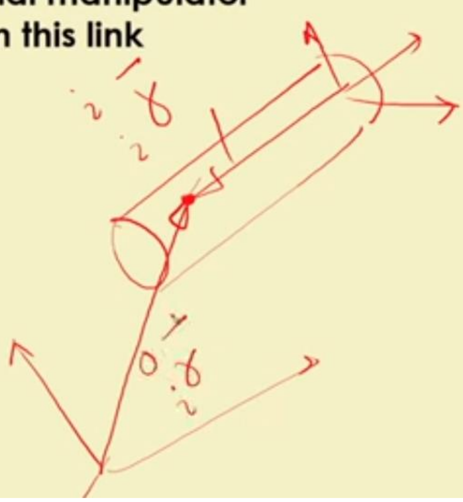
  - K: Total kinetic energy of robot
  - P: Total potential energy of robot
  - q_i: Joint variable of i-th joint
  - $\dot{q}_i$: first time derivative of q_i
  - τ_i : Generalized force (torque) at i-th joint

Let us consider *i*-th link of a serial manipulator
Position of a fixed point lying on this link

$$^{i}_{i}\bar{r} = \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

$$^{0}_{i}\bar{r} = {}^{0}_{i}T \; {}^{i}_{i}\bar{r}$$

where $^{0}_{i}T = {}^{0}_{1}T \; {}^{1}_{2}T \ldots\ldots {}^{i-1}_{i}T$



Inertia term

$$D_{ic} = \sum_{j=\max(i,c)}^{n} T_r\left(U_{jc} J_j U_{ji}^{T}\right) \qquad i,c = 1,2,\ldots,n$$

Coriolis and centrifugal term

$$h_{icd} = \sum_{j=\max(i,c,d)}^{n} T_r\left(U_{jcd} J_j U_{ji}^{T}\right) \qquad i,c,d = 1,2,\ldots,n$$

Gravity term

$$C_i = \sum_{j=i}^{n} \left(-m_j \bar{g} U_{ji} \, {}^{j}_{j}\bar{r}\right) \qquad i = 1,2,\ldots,n$$

$$\text{Let } \frac{\partial\, {}^{0}_{i}T}{\partial q_j} = U_{ij} \qquad \text{Therefore, } {}^{0}_{i}\bar{V} = \left(\sum_{j=1}^{i} U_{ij}\dot{q}_j\right) {}^{i}_{i}\bar{r}$$

Note: $$U_{ijk} = \frac{\partial U_{ij}}{\partial q_k}$$

Notes: To calculate the dynamic equation with this method we need to calculate:

- The inertia tensor of each link “J”.

- Then “U(jc)s”.

- The inertia term.

- The “U(jcd)s”.

- The Coriolis term.

- Find the “(j j)r”.

- The gravity term.

Tr: the trace of a matrix.