EE683 Robot Control: Assignment1

1st Donghee Han
Robotics Program
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
hdh7485@kaist.ac.kr

***Abstract*—In this paper, we explain the coordinate system, rotation and operator related to control and explain the results of the experiment of PD control in SE(3) first order system.**

***Index Terms*—Control, Robot**

I. Lecture 3

In the Lecture 3, we learned about the position of the rigid body and the method of rotational expression in the three-dimensional coordinate system. First, the position of the object is represented by the movement and rotation from the origin of the coordinate system, which is represented by r, R respectively. You can then use a vector to express the coordinates and speeds of the position from the frame and the origin of the different frame.

The rotation between the two frames can be interpreted in three ways. Can be used to move the coordinates of the vector from one frame. For example, vector defined ${}^1r_p$ in frame1 to convert frame2, It can calculate ${}^0r_p = {}^0R_1 {}^1p$. Rotation can be used as an operator to rotate vectors. For example, to rotate from (frame0) to (frame1), multiply the rotation matrix. Can be used to indicate the rotation of one frame to another.

The rotation matrix can be expressed in SO(3) (Special Orthogonal group). The group can do the binary operation $\cdot$ and it can follow the following rules:

```
1. Closure
2. Associativity
3. Identity
4. Inverse
```

$$SO(3) = \left\{R \in \mathbb{R}^{3\times3} R^T R = RR^T = I, \det(R) = +1\right\} \tag{1}$$

Homogeneous transformation means a transformation that moves the rotation and position together. To express p of A frame as B frame, rotate by multiplying the rotation matrix ${}^AR_B$, then add the distance between the origin of A frame and the origin of B frame. If this is expressed in a single matrix, it can be expressed as follows:

$$\begin{pmatrix} {}^Ar_p \\ 1 \end{pmatrix} = \begin{bmatrix} {}^AR_B & {}^Ar_B \\ 0 & 1 \end{bmatrix} \begin{pmatrix} {}^Br_p \\ 1 \end{pmatrix} \tag{2}$$

Assuming that know the Homogeneous transformation between A frame and B frame and B frame and C frame, It can be calculated the Homogeneous transformation of A frame and C frame by internalizing the two Homogeneous transformation. Homogeneous transformation can be expressed in SE (3) (Special Euclidean group). This group also follows the definition of the group described for the purpose.

Because the rotation matrix is Orthogonal matrix, the $RR^T = I, R^{-1} = R^T$ is established and the inverse of Homogeneous transformation is as follows.

$$T = \begin{bmatrix} R & r \\ 0 & 1 \end{bmatrix}, T - 1 = \begin{bmatrix} R^T & -R^T r \\ 0 & 1 \end{bmatrix} \tag{3}$$

In order to calculate Linear Velocity, calculate dot product between transpose of rotation matrix and time derivative of displacement vector. However, the speed of rotation matrix is different.

The three-dimensional skew-symmetric matrix can be expressed in three-dimensional vectors through vee and hat operators.

$$(R^T \dot{R})^\vee = \omega \tag{4}$$

Body twist is combination of linear velocity and angular velocity matrix $V = \begin{bmatrix} v \\ w \end{bmatrix}$. The result of body twist's hat operator is $T^{-1}\dot{T}$, and the result is $\begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix}$

If use dot matrix between body twist and adjoint matrix, it can calculate the body twist in another frame. Adjoint matrix has following properties.

$$\begin{aligned} {}^{A}Ad_{B}{}^{B}Ad_{C} &= {}^{A}Ad_{C} \\ {}^{A}Ad_{B}^{-1} &= {}^{B}Ad_{A} \\ {}^{A}T_{B} * V_{B}{}^{A}T_{B}^{-1} &= {}^{A}Ad_{B} * V_{B} = {}^{B}Ad_{A}^{-1} * V_{B} \end{aligned}$$

II. Lecture 4

There are many ways to indicate rotation. First of all, SO(3) is expressed in a matrix consisting of nine elements. This matrix is $R^T R = I$, and the sum of the squared elements is 1. It can express rotation using three parameters without using all nine elements.

Exponential coordinates can be expressed using a rotation axis ω and a rotation angle θ. Vector r consists of three parameters. The rotation axis vector is three-dimensional, so it consists of three elements and uses scalar θ to represent the angle of rotation. The reason for the exponential coordinates is that $e^{\omega\theta}$ equivalent with rotation matrix.

$$e^{\hat{\omega}\theta} = I + \sin\theta\hat{\omega} + (1-\cos\theta)\hat{\omega}^2 \in SO(3) \quad (5)$$

means $e^{\hat{\omega}\theta}$ is in the SO(3) groups and this formula call Rodrigues' formula.

$\omega\theta$ can be expressed using ξ, and $\|\xi\|$ is the result of normalize θ, ω. The above equation is formed by the following expression.

$$\hat{\xi} = I + \frac{\sin\|\xi\|}{\|\xi\|}\hat{\xi} + \frac{(1-\cos\|\xi\|)}{\|\xi\|^2}\hat{\xi}^2 \in SO(3) \quad (6)$$

The reasons for representing angular velocity as $R^T\dot{R} = \hat{\omega}$ are as follows: Each column of the rotation matrix represents the x, y, and z axes. This is because $\dot{R} = \hat{\omega}R$, and multiplying the equation by R^T results in $R^T\hat{\omega}$.

If you integrate the speed, you will find the position. So if you integrate the rotational speed, does the angle come out? Not really. Let's try to integrate the rotational speed. $R^T\dot{R} = \hat{\omega}$ can be expressed as $\dot{R} = R(t)\hat{\Omega}$, which in turn becomes

$$R(k+1) = R(k)exp(\Delta T\hat{\Omega}(k)) \quad (7)$$

Logarithmic operation of Rotation matrix is not a rotational speed. In order to get the rotational speed from the external codes, it need to do a dexp operation. Because $\hat{\omega} = R^T\dot{R}$, which is $= \mathrm{dexp}_{-\hat{\xi}}\dot{\widehat{\xi}}$ is derived.

The Euler angle defines the rotation angle in three rotations. The Euler angles have intrinsic and extrinsic rotations, and can be rotated in 12 order, but mainly zyz and XYZ are used. intrinsic zyx = extrinsic xyz = roll-pitch-yaw is widely used in aviation. zyz is used a lot in robots. The Euler angle is useful in the field of robots, but it has its disadvantage because of creating a gimbal lock.

Following example is a gimbal lock case. When the β angle is $\pm$ 90 degrees, the α and γ are not distinguished, and this is called gimbal lock or singularity. All rotation methods expressed in three parameters produce singularity.

Quaternion is a method of expressing SO(3) in four parameters. Quaternion consists of scalar part(q0) and vector part(q=(q1,q2,q3)T) and norm(q) is 1. This method of expression does not produce singularity and small angles are not expressed in small numbers. However, the ambiguity of the sign exists because four parameters, not the least, are not used and q = -q = R. The angular velocities and body angular velocities of roll pitch yaw are established as follows:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\psi\cos\phi - \cos\theta\sin\phi\sin\psi & \cos\psi\sin\phi + \cos\theta\cos\phi\sin\psi & \sin\theta\sin\psi \\ -\sin\psi\cos\phi - \cos\theta\sin\phi\cos\psi & -\sin\psi\sin\phi + \cos\theta\cos\phi\cos\psi & \sin\theta\cos\psi \\ \sin\phi\sin\theta & -\cos\phi\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (8)$$

Log operation to SE (3) will calculate the exponential coordinate, and if the exponential operation is performed again, SO(3) will be calculated. And if do $T^{-1}\dot{T}$ operation, can get body twist, and if do dexp operation, it can get time derivative result of exponential coordinate. It's the same concept as SE(3) but the equation becomes a little more complicated.

Screw motion is proposed by Chasles-Mozzi and can represent the position of any rigid body by rotation and movement from the screw shaft. And it can change the order of rotation and movement, and it is called screw motion to do it at the same time. When described in exonential and log map of SE(3) the $\xi\|\xi\|$ means the axis of rotation, and $\|\xi\|$ means the amount of rotation.

III. Lecture 5

Body wrench is the six-dimensional vector which consist of a linear force and a angular moment.

$$F = \begin{bmatrix} f & b \end{bmatrix} \in \mathbb{R}^6 \quad (9)$$

The inner production of body wrench and body twist means the power.

$$\dot{W} = f^T v + n^{\omega} \quad (10)$$

The formula above is used for converting the body wrench's frame.

$$\dot{W} = V_a^T F_a = V_B^T F_B F_a = {}^{A}\,\mathrm{Ad}_B^{-T}\, F_B \quad (11)$$

Different adjoin matrix is used to transform the coordinate system of Body twist and Body wrench.

To convert a frame of Force $F = M\dot{V} + CV + q$의 frame, use power $V = \mathrm{Ad}(V')$. From the result, time derivative of Adjoint transformation $ad = Ad_T^{-1}\dot{Ad}_T$ is generated. Body acceleration can be calculated using adjoint operator. When time derivative the body twist, the result is $\dot{V} = Ad^{-1}\dot{V} + \dot{V} - Ad^{-1}\dot{A}dAd^{-1}V_0$. If replace to adjoint operator, the result following $Ad^{-1}\dot{V} + \dot{V} - \mathrm{ad}\,\mathrm{Ad}^{-1}\,V_0$. Time derivative of body twist makes body acceleration but this does not mean physical linear, rotational acceleration vectors.

Adjoint opertaor, when inner product to Body twist, results as follows:

$$\begin{aligned}\mathrm{ad}_{V_1} V_k &= \begin{bmatrix} \widehat{w}_1 & \widehat{v}_1 \\ 0 & \widehat{w}_1 \end{bmatrix} \begin{bmatrix} v_k \\ w_k \end{bmatrix} \\ &= \begin{bmatrix} w_1 \times v_k + v_1 \times w_k \\ w_1 \times w_k \end{bmatrix}\end{aligned}$$

If $\mathrm{ad}_{V_1} V_1$, the result is following $w_1 \times v_1 - (w_1 \times v_1) = 0$.

The coordinate transformation of Body wrench and momentum uses the same Adjoint matrix. In terms of the concept of the tensor, the speed is the vector, and the force is the covector. The covector is a value that has a physical meaning with a vector and should be independent of the coordinate system. When converting vectors into another coordinates, use following equation.

$$V_B = {}^A\mathrm{Ad}_B^{-1} V_A$$

And to convert the frame of covector use following equation

$$F_B = {}^A\mathrm{Ad}_B^T F_A$$

Body twist and body wrench is vector and covector. So if established following equation, $F^T V = 0$ it called reciprocal, not orthogonal.

The angular velocity and linear velocity expressed in a fixed frame not body frame are called spatial twist.

When Kinematic control is performed, the input is three dimensional vector of the angular velocity is used. However, the error of rotation should be converted to an exponential coordinate by log operation. Same in SE (3) but with linear velocity.

IV. Lecture 6

A. Lie Group and Lie Algebra

Before explaining the Lie group, we need to know the Differentiable manifold. Differentiable manifold means treating the tangent space of any space or shape as an euclidean space. SO(3) and SE(3) are the Lie group. so(3) and se(3) are the tangent spaces of SO(3) and SE(3) called Lie algebra. And hat operation performed so(3) and se(3) are equivalent to $\mathbb{R}^3$ and $\mathbb{R}^6$.

B. PD Control

Before explaining PD control in Euclidean space, I will explain kinematic control in $\mathbb{R}^3$ first order system. In the situation of the Regulation, if insert the velocity into the Control input, the result will be v_b as it is entered. That is $v_b = u$. control law $u = R^T$ according to $\dot{r} + k_p\tilde{r} = 0$ is established. Control Raw is different when tracking. $u = R^T(\dot{r} + k_p(r_d - r)$ and the other is the same as Regulation. Therefore, in the close loop system, $\dot{\tilde{r}} + k_p\tilde{r} = 0$

To solve the kinematic equation In $\mathbb{R}^n$ second order system, we can use First control raw $\tau = M(C\dot{q} + K_d\dot{q} + K_p(q_d - q))$. If using second law, it can be solved easier than using first law. Second control raw $\tau = K_d\dot{q} + K_p(q_d - q)$ cancelation M and C because $\dot{M} - 2C$ is skew symmetric.

The systems used in the regulation kinematic control of SO(3) in the First order system are as follows:

$$\dot{R} = R\hat{U}, \omega = U \tag{12}$$

Control input U can be expressed as $-k_p \log(R)^\vee = -k_p\xi$, and control goal is to ensure that R_t is Identity Matrix. When Lyapunov method analyzed in Lie algebra, $V(\xi) = 1/2\xi^T\xi$ and time derivative of this results in $\dot{V}(\xi) = -k_p\|\xi\|^2$. As a result, the system becomes exponential external.

Tracking kinematic control is similar with regulation control.The Control Goal is to ensure that $R(t)$ and $\omega_b(t)$ are $R_d(t)$ and $V_d(t)$. Define the error of SO(3) $\tilde{R} = R_b^T R_d = {}^bR_d$ and control design is:

$$(\tilde{R^T}\dot{\tilde{R}})^\vee = \omega_d - {}^bR_d^T U \tag{13}$$

When organize expressions, get the following results:

$$(\tilde{R}^T\dot{\tilde{R}})^\vee = -k_p \log(\tilde{R})^\vee \tag{14}$$

As a result, control becomes exponential stable.

SO(3) The goal of the regulation PD control in the second order system is to ensure that R matrix be I. The equation of motion is $\dot{R} = Rom\tilde{e}ga_b$ and $\dot{\omega}_b = f(R, \omega_b) + U$. In second order SO(3) systems, PD Control's P action is $-k_p \log(R)$, and D action is $-k_d\omega_b$. As a result,

$$U = -f(R, \Omega_b) - K_p\xi - K_d\omega_b \tag{15}$$

and it has a Exponential stability.

We should to know that log map has a singularity. Result of $\frac{1}{2}\|\xi(t)\|^2 \leq V_0(t) \leq V_0()$, There isn't singularity in the $\|\xi(t)\| < \pi$. Another calculation method for second order system is remove the non linear funct Because $-\hat{I_b}\omega_b$ is skew symmetric matrix. And the re of control law is:

$$U = -k_p \xi - K_d \omega_b$$

Above we checked the external stability in Lie alge SO(3) and so(3) are external decays, but not in SE(3) se(3). Now let's look at PD Control in SE (3). Firs all, the primary system of SE(3) regulation control is primary system. $\dot{T} = T\hat{V}$ and control input is

$$U = -\begin{bmatrix} (k_w + k_v) I_3 & 0 \\ 0 & k_w I_3 \end{bmatrix} \lambda$$

and it has exponential stability. Another method is couple the SO(3) and $\mathbb{R}^3$. Linear velocity is expressed in $v_b = -R^T K_v r$, angular velocity is expressed in $\omega_b = -K_\omega \xi$. And each of them exponential stable in closed loop dynamics.

In secondary systems, SO(3) and $\mathbb{R}^3$ can also be calculated together or independently.

$$\begin{aligned} U &= -f(T, V_b) - K_p \log(T) - K_d V_b \\ U &= -f(T, V_b) - \begin{bmatrix} K_\omega \log(R) \\ R^T K_v r \end{bmatrix} - K_d V_b \end{aligned}$$

The exponential decay of se(3) does not equivalent with SE(3). The SO(3) is bounded so that it can exponential decay, but SE(3) is not external decay because it can be boundless and infinite for $\mathbb{R}^3$.

V. Simulation and Result

I carried out the practical training with what I learned in this lecture. The example is to control SO(3) but I succeed to implement the controller in SE(3) using Matlab 2019a. The source code is vast enough to be included in this paper and can be downloaded at the following link. https://github.com/hdh7485/EE683/blob/master/Assignment_1/SE3_loop.m

The start matrix is $T_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ and the target matrix is $T_d = \begin{bmatrix} 0 & -0.707 & 0.707 & 1 \\ 0 & 0.707 & 0.707 & 1 \\ -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

A total of 500 matrix was used as T_d by interpolation. The period of the loop used for control is 0.01 seconds and the gain values used are $K_v = 10.0, K_\omega = 5.0 and K_d = 0.1$. To facilitate calculation, R and r were calculated by decouple each.

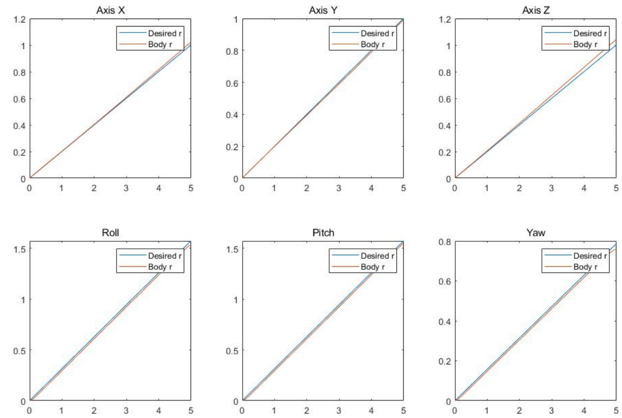

Fig. 1.

Simulation results show that the error converge close to zero but not equal to zero, but this is assumed to be steady-state error. It is thought to disappear if the gain value is raised or I control is added.