دانشگاه صنعتی خواجه نصیرالدین طوسی ۱۳۰۷

استاد: دکتر حمیدرضا تقی راد

بنام آنکه جان را فکرت آموخت

دانشکده مهندسی برق
گروه کنترل
نیمسال اول ۱۴۰۳– ۱۴۰۴

تاریخ: ۱۴۰۳/۰۸/۰۸

رباتیک

تمرین فصل چهارم

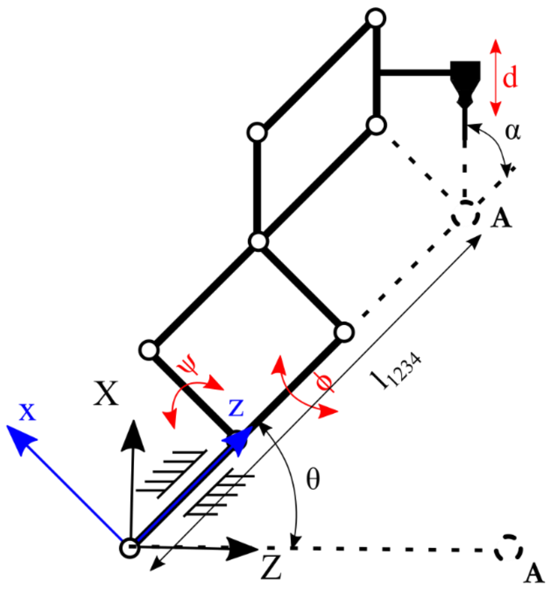
در مفاهیم رباتیکی، همواره نیاز است نقطه‌ای از یک بازوی رباتیک اندازه گیری شود تا از آن، در فیدبک کنترلی ربات استفاده شود. این نقطه سه بعدی فضایی به‌عنوان مثال، می‌تواند مجری نهایی ربات باشد. مختصات اندازه گیری این نقطه با $\boldsymbol{\chi} = [\boldsymbol{x}, \boldsymbol{\theta}]^T$ توصیف می‌شود و به آن، بردار مختصات کاری ربات گفته‌می‌شود. این بردار، که از 6 مولفه تشکیل شده‌است، خود از دو بخش بردار موقعیت و دوران به‌صورت زیر تشکیل‌می‌شود:

$$\boldsymbol{x} = [x, y, z]^T$$
$$\boldsymbol{\theta} = [\theta_x, \theta_y, \theta_z]^T$$

در این تمرین قصد داریم به محاسبه نرخ تغییرات موقعیت و جهت‌گیری مجری نهایی ربات به‌صورت زیر بپردازیم:

$$\dot{\boldsymbol{x}} = [\dot{x}, \dot{y}, \dot{z}]^T$$
$$\dot{\boldsymbol{\theta}} = [\dot{\theta}_x, \dot{\theta}_y, \dot{\theta}_z]^T$$

أ) ربات جراح چشم زیر را در نظر بگیرید. ربات فوق دارای حرکت دورانی حول محور z، دورانی حول محور y (یا Y) و حرکت خطی d است. سایر زوایا نظیر θ و α نیز ثابت است. اکنون می‌خواهیم ماتریس ژاکوبین ربات را محاسبه کنیم.



- با در نظر گرفتن فرمول سینماتیک مستقیم ربات، که در تمرین قبلی محاسبه شده است، ماتریس ژاکوبین ربات را به‌صورت $\boldsymbol{J} = \frac{\partial \boldsymbol{p}}{\partial \boldsymbol{q}}$ محاسبه کنید. توجه داشته باشید که متغیرهای مفصلی محرک ربات $\boldsymbol{q} = [\phi, \psi, d]^T$ است و بردار $\boldsymbol{p} = [x, y, z]^T$ سینماتیک مستقیم ربات است.

در این صورت به سادگی خواهیم داشت:

$$\mathbf{p} = \begin{bmatrix} sin(\theta)\left(l_{1234} + dcos(\alpha + \psi)\right) + dsin(\alpha + \psi)cos(\phi)cos(\theta) \\ d\,sin(\alpha + \psi)sin(\phi) \\ \cos(\theta)\left(l_{1234} + dcos(\alpha + \psi)\right) - dsin(\alpha + \psi)cos(\phi)sin(\theta) \end{bmatrix}$$

$$\boldsymbol{q} = [\phi, \psi, d]^T$$

پس با محاسبه ژاکوبین و ساده سازی های مثلثاتی خواهیم داشت:

$$\boldsymbol{J_v} = \frac{\partial \boldsymbol{p}}{\partial \boldsymbol{q}} = \begin{bmatrix} -dsin(\alpha + \psi)cos(\theta)sin(\phi) & dcos(\alpha + \psi)cos(\phi)cos(\theta) - dsin(\alpha + \psi)sin(\theta) & cos(\alpha + \psi) * sin(\theta) + sin(\alpha + \psi)cos(\phi)cos(\theta) \\ dsin(\alpha + \psi)cos(\phi) & dcos(\alpha + \psi)sin(\phi) & sin(\alpha + \psi)sin(\phi) \\ d * sin(\alpha + \psi)sin(\phi)sin(\theta) & -dsin(\alpha + \psi)cos(\theta) - dcos(\alpha + \psi)cos(\phi)sin(\theta) & cos(\alpha + \psi)cos(\theta) - sin(\alpha + \psi)cos(\phi)sin(\theta) \end{bmatrix}$$

برای کد نویسی در متلب نیز خواهیم داشت:

```
clear;
clc;

syms l_1234
syms theta alpha %passive joints
syms phi psi d %active joints

%% Forward Kinematics
P=simplify(Ry(theta)*Rz(phi)*(l_1234*[0;0;1]+d*Ry(psi+alpha)*[0;0;1]));
R=simplify(Ry(theta)*Rz(phi)*Ry(psi+alpha));

%% Jacobian
q=[phi,psi,d];
J_v=jacobian(P,q)
```

به تمام دانشجویان توصیه می شود که درباره چگونی محاسبه $\boldsymbol{J_\omega}$ از طریق روش بستار حلقه، تفکر کنند.

- با استفاده از ماتریس ژاکوبین محاسبه شده در بخش قبل، موقعیت‌های تکین ربات را محاسبه کنید.

$$\det(\boldsymbol{J_v}) = -d^2 sin(\alpha + \psi) \neq 0 \rightarrow \begin{cases} d \neq 0 \\ \psi + \alpha \neq k\pi \end{cases}$$

- {امتیازی} به‌صورت فیزیکی، موقعیت‌های تکین ذکر شده را توجیه کنید. افزون بر این، توضیح دهید چگونه یکی از موقعیت‌های فوق، می‌تواند در یک عمل جراحی کم تهاجمی، مفید باشد.

درباره نقطه دوران دور (RCM) و خواص آن تحقیق کنید و اینکه چگونه با هرگونه حرکت ربات، موقعیت نقطه مورد همواره ثابت خواهد ماند. از لینک های زیر برای فهم فیزیکی بیشتر استفاده کنید:

- https://youtu.be/P_f0BmmNTGc?si=LE9BPHiFIePnisvD
- https://youtu.be/8-Atao0Igyo?si=NGksjZC4Ev3spcnj
- https://youtu.be/G-wpTPNcUCs?si=uabpuj5rXG9IlP7e

ب) ربات صفحه‌ای PRR ذکر شده در مثال ۳,۳ کتاب را در نظر بگیرید. محور مختصات پایه را نظیر شکل ذکر شده در کتاب در نظر بگیرید.

- با استفاده از روش بستار حلقه، ماتریس ژاکوبین ربات را محاسبه کنید.

از قبل می دانستیم:

$$\mathbf{p} = \begin{bmatrix} d_1 + a_3cos(\theta_2 + \theta_3) + a_2cos(\theta_2) \\ a_3sin(\theta_2 + \theta_3) + a_2sin(\theta_2) \\ 0 \end{bmatrix}$$

$$\boldsymbol{q} = [d_1, \theta_2, \theta_3]^T$$

$$\boldsymbol{J_v} = \frac{\partial \boldsymbol{p}}{\partial \boldsymbol{q}} = \begin{bmatrix} 1 & -a3sin(\theta_2 + \theta_3) - a2sin(\theta_2) & -a_3sin(\theta_2 + \theta_3) \\ 0 & a3cos(\theta_2 + \theta_3) + a2cos(\theta_2) & a_3cos(\theta_2 + \theta_3) \\ 0 & 0 & 0 \end{bmatrix}$$

می توانید نظیر مثال ذکر شده در کتاب، ابتدا مستقیماً مشتق گیری کنید، سپس نتیجه را به صورت یک ماتریس مرتب کنید.

مجدداً به تمام دانشجویان توصیه می شود که درباره چگونی محاسبه $\boldsymbol{J_\omega}$ از طریق روش بستار حلقه، تفکر کنند.

- پاسخ خود را با استفاده از روش ژاکوبی مبتنی بر پیچه، صحت سنجی کنید.

ابتدا، می دانیم که ماتریس های دوران ربات به شرح زیر است:

$$\boldsymbol{R}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \boldsymbol{R}_2 = \begin{bmatrix} cos(\theta_2) & -sin(\theta_2) & 0 \\ sin(\theta_2) & cos(\theta_2) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{R}_3 = \begin{bmatrix} cos(\theta_2 + \theta_3) & -sin(\theta_2 + \theta_3) & 0 \\ sin(\theta_2 + \theta_3) & cos(\theta_2 + \theta_3) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

همچنین خواهیم داشت:

$$\boldsymbol{s}_3 = \boldsymbol{R}_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \boldsymbol{s}_2 = \boldsymbol{R}_2 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \boldsymbol{s}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \rightarrow Prismatic\ joint, Joint\ axis!$$

$$\boldsymbol{s}_{O_4} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \boldsymbol{s}_{O_3} = \boldsymbol{s}_{O_4} - \boldsymbol{R}_3 \begin{bmatrix} a_3 \\ 0 \\ 0 \end{bmatrix}, \boldsymbol{s}_{O_2} = \boldsymbol{s}_{O_3} - \boldsymbol{R}_2 \begin{bmatrix} a_2 \\ 0 \\ 0 \end{bmatrix}, \boldsymbol{s}_{O_1} = \boldsymbol{s}_{O_2} - \boldsymbol{R}_1 \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\$_1 = \begin{bmatrix} [0,0,0]^T \\ \boldsymbol{s}_1 \end{bmatrix}, \$_2 = \begin{bmatrix} \boldsymbol{s}_2 \\ \boldsymbol{s}_{O_2} \times \boldsymbol{s}_2 \end{bmatrix}, \$_3 = \begin{bmatrix} \boldsymbol{s}_3 \\ \boldsymbol{s}_{O_3} \times \boldsymbol{s}_3 \end{bmatrix}$$

$$\boldsymbol{J} = [\$_1 \quad \$_2 \quad \$_3] = \begin{bmatrix} \boldsymbol{J}_\omega \\ \boldsymbol{J}_v \end{bmatrix}$$

پس در نهایت خواهیم داشت:

$$\boldsymbol{J}_v = \begin{bmatrix} 1 & -a3sin(\theta_2 + \theta_3) - a2sin(\theta_2) & -a_3sin(\theta_2 + \theta_3) \\ 0 & a3cos(\theta_2 + \theta_3) + a2cos(\theta_2) & a_3cos(\theta_2 + \theta_3) \\ 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{J}_\omega = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

کد متلب زیر را نیز عمیقاً بررسی کنید. بررسی کنید زمان اجرای محاسباتی کدام روش محاسباتی بسیار کمتر است و چرا؟

```
clear;
clc;

%% Definition
syms d1 theta2 theta3 a2 a3;
theta=[d1;theta2;theta3]; %Joint variables!
Lengths=[0;a2;a3];         %Be careful about this!

%% Screw Table
S1=SR(1,0,0,0,0,0,0,theta(1));
S2=SR(0,0,1,0,0,0,theta(2),0);
S3=SR(0,0,1,Lengths(2),0,0,theta(3),0);
Screw=S1*simplify(S2*S3);

U0=[1;0;0];V0=[0;1;0];W0=[0;0;1];P0=[Lengths(2)+Lengths(3);0;0];Ze=[0,0,0];
S0=[U0,V0,W0,P0;Ze,1];

%% Screw Loop
tic
S1_Final=simplify(S1*S0);
[P_Total_S1,R_Total_S1]=PR_Link(S1_Final);
S2_Final=simplify(S1*S2*S0);
[P_Total_S2,R_Total_S2]=PR_Link(S2_Final);
S3_Final=simplify(S1*S2*S3*S0);
[P_Total_S3,R_Total_S3]=PR_Link(S3_Final);

P_End_Effector_Screw=P_Total_S3;
R_End_Effector_Screw=R_Total_S3;

%% Jacobian Analysis (Screw)
%Serious Notiec: Check page 215. Why didn't we use rotation matrices from
%DH?! Think about this.
tic
s3=R_Total_S2*[0;0;1];
s2=R_Total_S1*[0;0;1];
s1=[1;0;0];     %Notice! Prismatic Joint! Notice 2: Zi is the axis joint!
```

```
sO4=[0;0;0];
sO3=sO4-R_Total_S3*[a3;0;0];
sO2=sO3-R_Total_S2*[a2;0;0];
sO1=sO2-R_Total_S1*[0;0;0];

Do1=[[0;0;0];s1]; %Notice! Prismatic Joint!
Do2=[s2;cross(sO2,s2)];
Do3=[s3;cross(sO3,s3)];

J_Screw=[Do1,Do2,Do3];
Jv_Screw=J_Screw(4:6,:);
toc
Jw_Screw=J_Screw(1:3,:);


%% Loop closure Jacobian

tic
J_v=jacobian(P_End_Effector_Screw,theta);
toc

Error=simplify(Jv_Screw-J_v)
```

موفق باشید