دانشگاه خواجه نصیرالدین طوسی
دانشکده برق- گروه مکاترونیک

تمرین درس رباتیک دوره کارشناسی ارشد

رشته مهندسی مکاترونیک

عنوان

# تمرین رباتیک

نگارش

**علیرضا امیری**

مهر ۱۴۰۳

# فصل ۱

# پاسخ سوالات سری اول

## ۱.۱ توصیف جهت گیری

### ۱.۱.۱ پاسخ سوال ۱

### ۲.۱.۱ پاسخ سوال ۲

### ۳.۱.۱ پاسخ سوال ۳

وجود نویز در اندازه گیری ماتریس دوران، منجر به ذخیره ی اشتباه مقدار عددی بعضی از درایه ها خواهد شد. بنابراین، برای اصلاح ماتریس دوران، باید بر اساس یژگی های اساسی این ماتریس، مقادیر اشتباه را پیدا و تصحیح کرد. از جمله ی این ویژگی ها می توان به این موارد اشاره کرد:

۱. یکامتعامد

۲. دترمینان برابر ۱ باشد.

۳. مقدار وارون ماتریس با ترانهاده ی ماتریس برابر باشد.

۴. یک مقدار ویژه برابر ۱ داشته باشد.

### ۴.۱.۱ پاسخ سوال ۴

۱. در این بخش ابتدا یکه بودن هر یک از بردار های u، v و w را بررسی کرده و مقدار مجهول را با استفاده از این رابطه مشخص می کنیم.

$$۱ = \sqrt{x^{۲} + ۰٫۶۰۴۶^{۲} + ۰٫۰۹۷۷^{۲}} \Rightarrow \quad x = \pm ۰٫۷۹۰۵$$

$$۱ = \sqrt{۰٫۳۸۶۴^{۲} + ۰٫۳۶۸۶^{۲} + z^{۲}} \Rightarrow \quad z = \pm ۰٫۸۴۵۴$$

$$۱ = \sqrt{۰٫۴۷۵۲^{۲} + y^{۲} + ۰٫۵۲۵۰^{۲}} \Rightarrow \quad y = \pm ۰٫۷۰۶۰$$

با به دست آوردن مقادیر ممکن برای درایه های مجهول، لازم است علامت آنها نیز به درستی مشخص شود. برای این کار، با محاسبه ی دترمینان ماتریس دوران حاصل از هر یک از این پارامترها، می توانیم علامت صحیح پارامتر ها را در حالتی که دترمینان برابر ۱ باشد به دست آوریم.

```
1.3.1: Defining unknown elements in
  rotation matrix R

A = [0.7905, -0.3864, 0.4752;
0.6046, 0.3686, -0.7060;
0.0977, 0.8454, 0.5250]
det_A = det(A);

disp('Determinant of matrix A:'); disp(
  det_A)
```

در نتیجه:

$$x = +0/7905, \quad y = -0/7060, \quad z = +0/8454$$

- محاسبه زوایای اویلر حول محور ثابت:

```

1.3.2: Inverse Rpry



% Define the rotation matrix R with numerical
   values
disp(R)
% Calculate beta
beta = atan2(-R(3,1), sqrt(R(1,1)^2 + R(2,1)^2));

% Calculate gamma
gamma = atan2(R(2,1) / cos(beta), R(1,1) / cos(beta
   ));

% Calculate alpha
alpha = atan2(R(3,2) / cos(beta), R(3,3) / cos(beta
   ));

% Display the calculated angles in radians
fprintf('Calculated angles: alpha = %.2f rad, beta
   = %.2f rad, gamma = %.2f rad\n', alpha, beta,
   gamma);
```

```
% Optionally, convert radians to degrees
alpha_deg = rad2deg(alpha);
beta_deg = rad2deg(beta);
gamma_deg = rad2deg(gamma);

% Display the angles in degrees
fprintf('Calculated angles: alpha = %.2f deg, beta
   = %.2f deg, gamma = %.2f deg\n', alpha_deg,
   beta_deg, gamma_deg);
```

$$\alpha = ۵۸/۱۶^\circ, \quad \beta = -۵/۶۱^\circ, \quad \gamma = +۳۷/۴۱^\circ$$

- محاسبه زوایای اویلر حول محور متحرک :Ruvw

```
1.3.3: Inversw Ruvw
We calculate Euler angles (alpha, beta,
   gamma) from a given rotation matrix R.

% Define the rotation matrix R with
   numerical values
disp(R)
% Calculate beta
beta = atan2(R(1,3), sqrt(R(1,1)^2 + R(1,2)
   ^2));
```

```matlab

% Calculate alpha
alpha = atan2(-R(2,3) / cos(beta), R(3,3) / cos(beta));

% Calculate gamma
gamma = atan2(-R(1,2) / cos(beta), R(1,1) / cos(beta));

% Display the calculated angles in radians
fprintf('Calculated angles: alpha = %.2f rad, beta = %.2f rad, gamma = %.2f rad\n', alpha, beta, gamma);

% Optionally, convert radians to degrees
alpha_deg = rad2deg(alpha);
beta_deg = rad2deg(beta);
gamma_deg = rad2deg(gamma);

% Display the angles in degrees
fprintf('Calculated angles: alpha = %.2f deg, beta = %.2f deg, gamma = %.2f deg\n', alpha_deg, beta_deg, gamma_deg);
```

$$\alpha = ۵۳/۳۶^\circ, \quad \beta = ۲۸/۳۷^\circ, \quad \gamma = ۲۶/۰۵^\circ$$

- محاسبه زوایای اویلر حول محور متحرک :Rwvw

```
1.3.4: Inverse Rwvw
In this section we calculate the Euler
   angels given a Rotation Matrix in the
   system of wvw.
% Define the rotation matrix R with
   numerical values
disp(R)

% Calculate beta
beta = atan2(sqrt(R(3,1)^2 + R(3,2)^2),R
   (3,3));

% Calculate alpha
alpha = atan2(R(2,3) / sin(beta), R(1,3) /
   sin(beta));

% Calculate gamma
gamma = atan2(R(3,2) / sin(beta), -R(3,1) /
    sin(beta));

% Display the calculated angles in radians
fprintf('Calculated angles: alpha = %.2f
   rad, beta = %.2f rad, gamma = %.2f rad\n
   ', alpha, beta, gamma);
```

```
% Optionally, convert radians to degrees
alpha_deg = rad2deg(alpha);
beta_deg = rad2deg(beta);
gamma_deg = rad2deg(gamma);

% Display the angles in degrees
fprintf('Calculated angles: alpha = %.2f deg, beta = %.2f deg, gamma = %.2f deg\n', alpha_deg, beta_deg, gamma_deg);
```

$$\alpha = -۵۶/۰۶°, \quad \beta = ۵۸/۳۳°, \quad \gamma = ۹۶/۵۹°$$

- محاسبه زوایای اویلر حول محور متحرک :Rwuw

```

1.3.5: Inverse Rwuw
In this section we calculate the Euler angels given a Rotation Matrix in the system of wuw.
% Define the rotation matrix R with numerical values
disp(R)
% Calculate beta
beta = atan2(sqrt(R(3,1)^2 + R(3,2)^2),R(3,3));

% Calculate alpha
```

```
alpha = atan2(R(1,3) / sin(beta), -R(2,3) / sin(
   beta));

% Calculate gamma
gamma = atan2(R(3,1) / sin(beta), R(3,2) / sin(beta
   ));

% Display the calculated angles in radians
fprintf('Calculated angles: alpha = %.2f rad, beta
   = %.2f rad, gamma = %.2f rad\n', alpha, beta,
   gamma);

% Optionally, convert radians to degrees
alpha_deg = rad2deg(alpha);
beta_deg = rad2deg(beta);
gamma_deg = rad2deg(gamma);

% Display the angles in degrees
fprintf('Calculated angles: alpha = %.2f deg, beta
   = %.2f deg, gamma = %.2f deg\n', alpha_deg,
   beta_deg, gamma_deg);
```

$$\alpha = ۳۳٫۹۴^\circ, \quad \beta = ۵۸٫۳۳^\circ, \quad \gamma = ۶٫۵۹^\circ$$

۲. محاسبه ی زاویه و محور دوران معادل:

```
1.3.6: Inverse Screw
% Step 1: Define the rotation matrix R (example)
disp(R)

% Step 2: Calculate the rotation angle theta using the
    trace of R
theta = acos((trace(R) - 1) / 2);

% Step 3: Check if sin(theta) is zero (to avoid
    division by zero)
if sin(theta) == 0
error('Singularity encountered: sin(theta) is zero.
    Unable to compute the axis.');
end

% Step 4: Calculate the components of the screw axis s
    = [sx, sy, sz]
sx = (R(3,2) - R(2,3)) / (2 * sin(theta));
sy = (R(1,3) - R(3,1)) / (2 * sin(theta));
sz = (R(2,1) - R(1,2)) / (2 * sin(theta));

% Step 5: Display the calculated screw axis and
    rotation angle
fprintf('Calculated rotation angle (theta) in radians:
```

```
%.4f\n', theta);
fprintf('Calculated rotation angle (theta) in degrees: %.4f\n', rad2deg(theta));
fprintf('Screw axis vector s = [sx, sy, sz]: [%.4f, %.4f, %.4f]\n', sx, sy, sz);

% Optional: Normalize the screw axis (if you want the unit vector)
s_magnitude = sqrt(sx^2 + sy^2 + sz^2);
s_normalized = [sx, sy, sz] / s_magnitude;
fprintf('Normalized screw axis vector s: [%.4f, %.4f, %.4f]\n', s_normalized(1), s_normalized(2), s_normalized(3));

%% Step 6: Calculate the theta vector (theta_x, theta_y, theta_z)
theta_x = theta * sx;
theta_y = theta * sy;
theta_z = theta * sz;

% Step 7: Display the theta vector
fprintf('Theta vector = [theta_x, theta_y, theta_z]: [%.4f, %.4f, %.4f]\n', theta_x, theta_y, theta_z);
```

$$\theta = ۳۰/۰۰^\circ,$$

$$\mathbf{S} = [s_x, s_y, s_z] : [۰/۸۲۵۵, ۰/۲۰۰۹, ۰/۵۲۷۳]$$

$$\vec{\theta} = [\theta_x, \theta_y, \theta_z] : [۱/۰۰۸۵, ۰/۲۴۵۴, ۰/۶۴۴۲]$$

۳. محاسبه نمایش چهارگان ماتریس دوران $\mathbf{R}$:

```
% Step 1: Define the rotation matrix R (example)
disp(R)

% Step 2: Calculate the scalar part of the quaternion (e4)
e4 = 0.5 * sqrt(1 + R(1,1) + R(2,2) + R(3,3));

% Step 3: Calculate the vector components of the quaternion
e1 = (R(3,2) - R(2,3)) / (4 * e4);    % e1
e2 = (R(1,3) - R(3,1)) / (4 * e4);    % e2
e3 = (R(2,1) - R(1,2)) / (4 * e4);    % e3

% Step 4: Display the quaternion as a vector [e1, e2, e3, e4]
quaternion_from_R = [e1, e2, e3, e4];
disp('Quaternion vector [e1, e2, e3, e4]:');
disp(quaternion_from_R);
```

$$\vec{\epsilon} = [\epsilon_۱, \epsilon_۲, \epsilon_۳, \epsilon_۴] = [۰/۴۷۳۵, ۰/۱۱۵۲, ۰/۳۰۲۴, ۰/۸۱۹۲]$$

### ۵.۱.۱ پاسخ سوال ۵

با در اختیار داشتن ماتریس دوران، میتوان با استفاده از ویژگی های مختلف این ماتریس اقدام به پیدا کردن المان های مجهول کرد. یکی از راه ها، محاسبه ی ترانهاده و وارون ماتریس و برابر قرار دادن المان های متناظر است که به دلیل پیچیدگی زیاد این روش، از ویژگی دیگری استفاده می شود. به عنوان جایگزین، با استفاده از روابط ضرب خارجی ستون های ماتریس دوران چنان که در بخش پایین قرار داده شده است استفاده می کنیم.

$$\begin{aligned} \mathbf{u} \times \mathbf{v} &= \mathbf{w}, \\ \mathbf{v} \times \mathbf{w} &= \mathbf{u}, \\ \mathbf{w} \times \mathbf{u} &= \mathbf{v} \end{aligned} \tag{۱.۱}$$

ماتریس R مطابق صورت سوال به شکل زیر تعریف می شود:

$$R = \begin{pmatrix} \cos(t) & a & b \\ \sin(t) & \frac{\sqrt{2}\,k\,\cos(t)}{2} & c \\ 0 & -\frac{\sqrt{2}\,k\,\sin(t)}{2} & d \end{pmatrix}$$

با قرار دادن ستون های این ماتریس به عنوان u، v w و محاسبه ی ضرب خارجی آنها خواهیم داشت:

$$\mathbf{u} \times \mathbf{v} = \begin{pmatrix} -\frac{\sqrt{2}\,k\,\sin(t)^2}{2} \\ \frac{\sqrt{2}\,k\,\sin(2\,t)}{4} \\ -\frac{\sqrt{2}\,k\,\sin(t)^2}{2} - a\,\sin(t) + \frac{\sqrt{2}\,k}{2} \end{pmatrix}$$

$$\mathbf{v} \times \mathbf{w} = \begin{pmatrix} \frac{\sqrt{2}\,k\,(d\,\cos(t)+c\,\sin(t))}{2} \\ -a\,d - \frac{\sqrt{2}\,b\,k\,\sin(t)}{2} \\ a\,c - \frac{\sqrt{2}\,b\,k\,\cos(t)}{2} \end{pmatrix}$$

$$\mathbf{w} \times \mathbf{u} = \begin{pmatrix} -d\,\sin(t) \\ d\,\cos(t) \\ b\,\sin(t) - c\,\cos(t) \end{pmatrix}$$

با برابر قرار دادن این ماتریس ها با ستون های ماتریس دوران و حل معادلات به دست آمده به ازای مقادیر a،b،c،d،k به جواب های زیر دست می یابیم.

$$\mathbf{a} = \begin{pmatrix} -\frac{\sin(t)}{\sqrt{\sin(t)^{۲}+۱}} \\ \frac{\sin(t)}{\sqrt{\sin(t)^{۲}+۱}} \end{pmatrix}$$

$$\mathbf{b} = \begin{pmatrix} -\frac{\sin(t)^{۲}}{\sqrt{\sin(t)^{۲}+۱}} \\ \frac{\sin(t)^{۲}}{\sqrt{\sin(t)^{۲}+۱}} \end{pmatrix}$$

$$\mathbf{c} = \begin{pmatrix} \frac{\sin(۲\,t)}{۲\sqrt{\sin(t)^{۲}+۱}} \\ -\frac{\sin(۲\,t)}{۲\sqrt{\sin(t)^{۲}+۱}} \end{pmatrix}$$

$$\mathbf{d} = \begin{pmatrix} \frac{۱}{\sqrt{\sin(t)^{۲}+۱}} \\ -\frac{۱}{\sqrt{\sin(t)^{۲}+۱}} \end{pmatrix}$$

$$\mathbf{k} = \begin{pmatrix} \frac{\sqrt{۲}}{\sqrt{\sin(t)^{۲}+۱}} \\ -\frac{\sqrt{۲}}{\sqrt{\sin(t)^{۲}+۱}} \end{pmatrix}$$

### ۶.۱.۱ پاسخ سوال ۶

با در اختیار داشتن مقادیر زوایای اویلر، می توان ماتریس دوران و مشتق آن را به دست آورده و با استفاده از المان های این ماتریس ها، سرعت زاویه ای و ماتریس تبدیل آن را به دست آورد. برای این منظور، مطابق فرایند زیر عمل می کنیم.

```
%% Section 1: Define the Euler Angles and Rotation Matrices for UVW
% Define symbolic angles alpha, beta, gamma for the Euler angles
syms alpha beta gamma real
syms alpha_dot beta_dot gamma_dot real  % Time derivatives of the angles

% Step 1: Define symbolic rotation matrices for each axis
% Rotation about the x-axis (Ru(alpha))
Ru_alpha = [1,          0,           0;
0, cos(alpha), -sin(alpha);
0, sin(alpha),  cos(alpha)];

% Rotation about the y-axis (Rv(beta))
Rv_beta = [cos(beta),  0, sin(beta);
0,          1, 0;
-sin(beta), 0, cos(beta)];

% Rotation about the z-axis (Rw(gamma))
Rw_gamma = [cos(gamma), -sin(gamma), 0;
sin(gamma),  cos(gamma), 0;
0,           0,          1];
```

$$R_{u_\alpha} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$R_{v_\beta} = \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix}$$

$$R_{w_\gamma} = \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

آنگاه ماتریس دوران حاصل از سه دوران حول محور های اویلر از ضرب سه ماتریس فوق به دست می آید:

$$R_{uvw} = \begin{pmatrix} \cos(\beta)\,\cos(\gamma) & -\cos(\beta)\,\sin(\gamma) & \sin(\beta) \\ \cos(\alpha)\,\sin(\gamma) + \cos(\gamma)\,\sin(\alpha)\,\sin(\beta) & \cos(\alpha)\,\cos(\gamma) - \sin(\alpha)\,\sin(\beta)\,\sin(\gamma) & -\cos(\beta)\,\sin(\alpha) \\ \sin(\alpha)\,\sin(\gamma) - \cos(\alpha)\,\cos(\gamma)\,\sin(\beta) & \cos(\gamma)\,\sin(\alpha) + \cos(\alpha)\,\sin(\beta)\,\sin(\gamma) & \cos(\alpha)\,\cos(\beta) \end{pmatrix}$$

در ادامه و با محاسبه ی مشتق این ماتریس، سرعت زاویه ای $\omega$ را محاسبه می کنیم.

```
%% Section 1: Define the Euler Angles and Rotation Matrices for UVW
% Define symbolic angles alpha, beta, gamma for the Euler angles
syms alpha beta gamma real
syms alpha_dot beta_dot gamma_dot real % Time derivatives of the angles

```

```
% Step 1: Define symbolic rotation matrices
    for each axis
% Rotation about the x-axis (Ru(alpha))
Ru_alpha = [1,          0,             0;
0, cos(alpha), -sin(alpha);
0, sin(alpha),  cos(alpha)];

% Rotation about the y-axis (Rv(beta))
Rv_beta = [cos(beta),  0, sin(beta);
0,          1, 0;
-sin(beta), 0, cos(beta)];

% Rotation about the z-axis (Rw(gamma))
Rw_gamma = [cos(gamma), -sin(gamma), 0;
sin(gamma),  cos(gamma), 0;
0,           0,           1];
```

$$\begin{pmatrix} -\dot{\beta}\,\cos(\gamma)\,\sin(\beta)-\dot{\gamma}\,\cos(\beta)\,\sin(\gamma) & \dot{\beta}\,\sin(\beta)\,\sin(\gamma)-\dot{\gamma}\,\cos(\beta)\,\cos(\gamma) & \dot{\beta}\,\cos(\beta) \\ \dot{\gamma}\,\sigma_2-\dot{\alpha}\,\sigma_1+\dot{\beta}\,\cos(\beta)\,\cos(\gamma)\,\sin(\alpha) & -\dot{\alpha}\,\sigma_3-\dot{\gamma}\,\sigma_4-\dot{\beta}\,\cos(\beta)\,\sin(\alpha)\,\sin(\gamma) & \dot{\beta}\,\sin(\alpha)\,\sin(\beta)-\dot{\alpha}\,\cos(\alpha)\,\cos(\beta) \\ \dot{\alpha}\,\sigma_4+\dot{\gamma}\,\sigma_3-\dot{\beta}\,\cos(\alpha)\,\cos(\beta)\,\cos(\gamma) & \dot{\alpha}\,\sigma_2-\dot{\gamma}\,\sigma_1+\dot{\beta}\,\cos(\alpha)\,\cos(\beta)\,\sin(\gamma) & -\dot{\alpha}\,\cos(\beta)\,\sin(\alpha)-\dot{\beta}\,\cos(\alpha)\,\sin(\beta) \end{pmatrix}$$

where

$R_{uvw\cdot} =$ $\sigma_1 = \sin(\alpha)\,\sin(\gamma)-\cos(\alpha)\,\cos(\gamma)\,\sin(\beta)$

$\sigma_2 = \cos(\alpha)\,\cos(\gamma)-\sin(\alpha)\,\sin(\beta)\,\sin(\gamma)$

$\sigma_3 = \cos(\gamma)\,\sin(\alpha)+\cos(\alpha)\,\sin(\beta)\,\sin(\gamma)$

$\sigma_4 = \cos(\alpha)\,\sin(\gamma)+\cos(\gamma)\,\sin(\alpha)\,\sin(\beta)$

$$\Omega = \begin{pmatrix} \dot{\alpha}+\dot{\gamma}\,\sin(\beta) \\ \dot{\beta}\,\cos(\alpha)-\dot{\gamma}\,\cos(\beta)\,\sin(\alpha) \\ \dot{\beta}\,\sin(\alpha)+\dot{\gamma}\,\cos(\alpha)\,\cos(\beta) \end{pmatrix}$$

در گام آخر، ماتریس تبدیل E با استفاده از تبدیل ژاکوبین و کمی ساده سازی به شکل زیر به دست می آید:

```
% Construct the matrix E by taking the Jacobian of Omega
    with respect to [alpha_dot; beta_dot; gamma_dot]
E_matrix = jacobian(Omega_vec, [alpha_dot, beta_dot,
    gamma_dot]);

% Simplify the matrix E
E_matrix = simplify(E_matrix);

% Display the matrix E(alpha, beta, gamma)
disp('Matrix E(alpha, beta, gamma):');
```

```
disp(E_matrix);
```

$$E = \begin{pmatrix} ۱ & ۰ & \sin\left(\beta\right) \\ ۰ & \cos\left(\alpha\right) & -\cos\left(\beta\right)\,\sin\left(\alpha\right) \\ ۰ & \sin\left(\alpha\right) & \cos\left(\alpha\right)\,\cos\left(\beta\right) \end{pmatrix}$$

### ۷.۱.۱ پاسخ سوال ۷