Robotics Project1

0860077 王國倫 電控所碩一

- 介面說明

  本次專題使用 MATLAB 做開發，使用前須輸入指定的數字(1~3)，即可開始使用相對應的任務。

  1.Forward Kinematic

```
>> project1
1.Forward kinematic
2.Inverse kinematic
3.exit
Please input (1~3):
1

input six joint variables:[theta1,theta2,theta3,theta4,theta5,theta6]
[50,50,50,50,50,50]
(n a o p)
   -0.8955    0.4342   -0.0976    0.3407
    0.1912    0.5734    0.7966    0.6378
    0.4019    0.6947   -0.5965   -0.3864
         0         0         0    1.0000

Cartesian point
(x,y,z,phi,theta,phi) = (0.340682 ,0.637812 ,-0.386425 , 96.984605 ,126.622640 ,120.047287 )
```

  2.Inverse Kinematic

```
Cartesian point
(x,y,z,phi,theta,phi) = (0.340682 ,0.637812 ,-0.386425 , 96.984605 ,126.622640 ,120.047287 )
>> project1
1.Forward kinematic
2.Inverse kinematic
3.exit
Please input (1~3):
2

T=[nx ox ax px ;  ny oy ay py ; nz oz az pz ;0 0 0 1;]
 [-0.8955    0.4342   -0.0976    0.3407;
    0.1912    0.5734    0.7966    0.6378;
    0.4019    0.6947   -0.5965   -0.3864;
         0         0         0    1.0000;]
(θ1,θ2,θ3,θ4,θ5,θ6) =
   49.9983   50.0011   49.9949   49.9995   50.0061   50.0007

(θ1,θ2,θ3,θ4,θ5,θ6) =
   49.9983   50.0011   49.9949 -130.0005  -50.0061 -129.9993

(θ1,θ2,θ3,θ4,θ5,θ6) =
theta2 is out of range!
 -106.2187 -187.2759   49.9949  -66.5344   20.1726   -9.1126
```

```
(θ1,θ2,θ3,θ4,θ5,θ6) =
theta2 is out of range!
 -106.2187 -187.2759   49.9949  113.4656  -20.1726  170.8874

(θ1,θ2,θ3,θ4,θ5,θ6) =
theta3 is out of range!
   49.9983    7.2759  135.2943   97.0145   36.2514  -11.2203

(θ1,θ2,θ3,θ4,θ5,θ6) =
theta3 is out of range!
   49.9983    7.2759  135.2943  -82.9855  -36.2514  168.7797

(θ1,θ2,θ3,θ4,θ5,θ6) =
theta2 is out of range!
theta3 is out of range!
theta4 is out of range!
 -106.2187 -230.0011  135.2943 -149.3660   38.3628   80.7941

(θ1,θ2,θ3,θ4,θ5,θ6) =
theta2 is out of range!
theta3 is out of range!
 -106.2187 -230.0011  135.2943   30.6340  -38.3628  -99.2059
```

3.Exit

```
>> project1
1.Forward kinematic
2.Inverse kinematic
3.exit
Please input (1~3):
3
exit!
```

- 程式架構說明

  等待使用者輸入數字，並執行相對應的任務。設立角度和弧度之間的換算，角度制轉成弧度制(degree_to_rad)和弧度制轉成角度制(rad_to_degree)，設定一個 valid 的變數來儲存輸入的角度是否在範圍內，並在後續進行判斷。

```
number = input('1.Forward kinematic\n2.Inverse kinematic\n3.exit\nPlease input (1~3):');

degree_to_rad = pi/180;  %transfer degree to rad
rad_to_degree = 180/pi;  %transfer rad to degree
valid = true;  %make sure every input joint in the range
```

1.Forward kinematic

當輸入數字為 1 時，執行順向運動學的計算，根據 DH model 的參數表訂出相關的變數設置，接著等待使用者輸入各個軸的角度，並轉換成弧度制做計算。

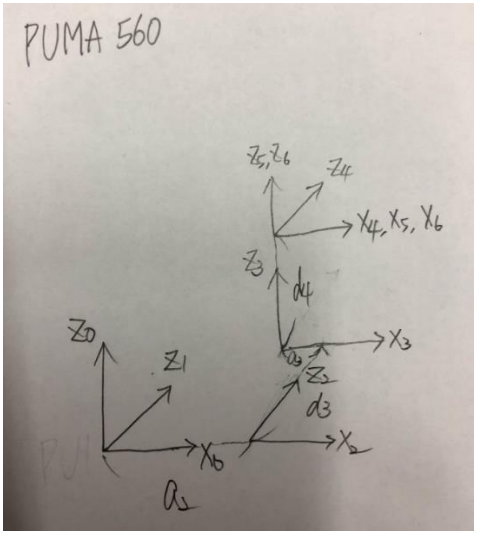
```
%Forward kinematic
if number == 1

    %kinematic table parameter
    a2 = 0.432;
    a3 = -0.02;
    d3 = 0.149;
    d4 = 0.433;

    %input six joint variables
    theta = input('\ninput six joint variables:[theta1,theta2,theta3,theta4,theta5,theta6]\n');
    theta1=theta(1)*degree_to_rad;
    theta2=theta(2)*degree_to_rad;
    theta3=theta(3)*degree_to_rad;
    theta4=theta(4)*degree_to_rad;
    theta5=theta(5)*degree_to_rad;
    theta6=theta(6)*degree_to_rad;
```

DH model 和各個軸的限制角度範圍

Joint	d(m)	a(m)	α	θ
1	0	0	-90°	0°
2	0	0.432	0°	0°
3	0.149	-0.02	90°	0°
4	0.433	0	-90°	0°
5	0	0	90°	0°
6	0	0	0°	0°



$$-160° \le \theta_1 \le 160°,\ -125° \le \theta_2 \le 125°$$
$$-135° \le \theta_3 \le 135°,\ -140° \le \theta_4 \le 140°$$
$$-100° \le \theta_5 \le 100°,\ -260° \le \theta_6 \le 260°$$

接收使用者回傳的值後，先確認角度是否有在符合的範圍之內。

```
% verify every joint angle
while abs(theta(1))>160
    fprintf('Theat1 out of range\n');
    valid =false;
    break
end

while abs(theta(2))>125
    fprintf('Theat1 out of range\n');
    valid =false;
    break
end

while abs(theta(3))>135
    fprintf('Theat1 out of range\n');
    valid =false;
    break
end

while abs(theta(4))>140
    fprintf('Theat1 out of range\n');
    valid =false;
    break
end

while abs(theta(5))>100
    fprintf('Theat1 out of range\n');
    valid =false;
    break
end

while abs(theta(6))>260
    fprintf('Theat6 out of range\n');
    valid =false;
    break
end
```

若超出範圍內將不列入計算。

```
>> project1
1.Forward kinematic
2.Inverse kinematic
3.exit
Please input (1~3):
1

input six joint variables:[theta1,theta2,theta3,theta4,theta5,theta6]
[300,50,50,50,50,50]
Theat1 out of range
error!
```

當各軸的角度都在其範圍內，進行順向順動學的運算，得出轉換的矩陣和 Cartesian 座標的點，並將結果進行輸出。關於數學的運算部份將和下一個數學運算的章節說明一起講解。

```
% output result
disp('(n a o p)');
disp(T6);

disp('Cartesian point')
fprintf('(x,y,z,phi,theta,phi) = (%f ,%f ,%f , %f ,%f ,%f )\n',px,py,pz,phi,theta,psi)
```

2.Inverse kinematic

當輸入為 2 時，執行逆向運動學的計算，並等待使用者輸入轉移矩陣，輸入完畢後將計算所有符合該矩陣的各軸角度，數計計算部份也一並在數學運算說明中解講。

```
%Inverse Kinematic
elseif number == 2
    % input n o a p transformation matrix
    T=input('\nT=[nx ox ax px ;  ny oy ay py ; nz oz az pz ;0 0 0 1;]\n');
```

當各軸角度運算完後，會執行一個函式，將所有得出的弧度轉換角度制，同時檢查該角度是否符合剛開始的角度限制，並將結果進行輸出。

```
% transfer rad to degree, verify its angle limitation, and output result
function r2a(theta1,theta2,theta3,theta4,theta5,theta6)

        rad_to_degree = 180/pi;  %transfer rad to degree

        % transfer every joint to degree
        angle=[];
        angle(1)=theta1*rad_to_degree;
        angle(2)=theta2*rad_to_degree;
        angle(3)=theta3*rad_to_degree;
        angle(4)=theta4*rad_to_degree;
        angle(5)=theta5*rad_to_degree;
        angle(6)=theta6*rad_to_degree;

        disp('(θ1,θ2,θ3,θ4,θ5,θ6) =')

        % make sure joint degree in the range
        if abs(angle(1))>160
            fprintf("theta1 is out of range!\n");
        end
        if abs(angle(2))>125
            fprintf("theta2 is out of range!\n");
        end
        if abs(angle(3))>135
            fprintf("theta3 is out of range!\n");
        end
        if abs(angle(4))>140
            fprintf("theta4 is out of range!\n");
        end
        if abs(angle(5))>100
            fprintf("theta5 is out of range!\n");
        end
        if abs(angle(6))>260
            fprintf("theta6 is out of range!\n");
        end

        % output result
        disp([angle(1),angle(2),angle(3),angle(4),angle(5),angle(6)]);
end
```

- 數學運算說明

  1.Forward kinematic

  順向運動學的運算，根據 DH model 和公式(1)，將各軸之間的轉換矩陣列出來。

$$A_n = Rot(z,\theta_n)*Trans(0,0,d_n)*Trans(a_n,0,0)*Rot(x,\alpha_n)$$

$$= \begin{pmatrix} c\theta_n & -s\theta_n c\alpha_n & s\theta_n s\alpha_n & a_n c\theta_n \\ s\theta_n & c\theta_n c\alpha_n & -c\theta_n s\alpha_n & a_n s\theta_n \\ 0 & s\alpha_n & c\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \ldots(1)$$

```
%transformation matrix A1~A6
A1 = [ cos(theta1)   0    -sin(theta1)    0;
       sin(theta1)   0     cos(theta1)    0;
                 0  -1              0     0;
                 0   0              0     1  ];

A2 = [ cos(theta2)   -sin(theta2)    0    a2*cos(theta2);
       sin(theta2)    cos(theta2)    0    a2*sin(theta2);
                 0              0    1                 0;
                 0              0    0                 1  ];

A3 = [ cos(theta3)   0     sin(theta3)   a3*cos(theta3);
       sin(theta3)   0    -cos(theta3)   a3*sin(theta3);
                 0   1              0               d3;
                 0   0              0                1  ];

    A4 = [ cos(theta4)     0    -sin(theta4)     0;
           sin(theta4)     0     cos(theta4)     0;
                     0    -1              0     d4;
                     0     0              0      1  ];

    A5 = [ cos(theta5)     0     sin(theta5)      0;
           sin(theta5)     0    -cos(theta5)      0;
                     0     1              0       0;
                     0     0              0       1  ];

    A6 = [ cos(theta6)   -sin(theta6)     0     0;
           sin(theta6)    cos(theta6)     0     0;
                     0              0     1     0;
                     0              0     0     1  ];
```

將各軸間的轉換矩陣相乘，得出最後的轉換矩陣和 n、o、a 和 p 的值。

```
% calculate final transformation matrix
T6=A1*A2*A3*A4*A5*A6;

% get n o a p value
nx = T6(1,1);
ny = T6(2,1);
nz = T6(3,1);

ox = T6(1,2);
oy = T6(2,2);
oz = T6(3,2);

ax = T6(1,3);
ay = T6(2,3);
az = T6(3,3);

px = T6(1,4);
py = T6(2,4);
pz = T6(3,4);
```

利用轉換矩陣得出ϕ, θ, ψ，並根據 Euler angle 方式來進行計算，其算式可參考公式(2)~(4)，最後將轉換矩陣和$(x, y, z, \phi, \theta, \psi)$進行輸出。

$$Euler(\phi,\theta,\psi) = \mathrm{T}$$

$$= \begin{pmatrix} \cos\phi\cos\theta\cos\psi - \sin\phi\sin\psi & -\cos\phi\cos\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\sin\theta & 0 \\ \sin\phi\cos\theta\cos\psi + \cos\phi\sin\psi & -\sin\phi\cos\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\sin\theta & 0 \\ -\sin\theta\cos\psi & \sin\theta\sin\psi & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$if\ \theta \neq 0, \qquad \phi = tan^{-1}\frac{a_y}{a_x} \quad (2)$$

$$\theta = tan^{-1}\frac{\sqrt{a_y^2 + a_x^2}}{a_z} \quad (3)$$

$$if\ \theta \neq 0, \qquad \psi = tan^{-1}\frac{o_z}{-n_z} \quad (4)$$

```
% transfer n o a to phi theta psi
phi=atan2(ay,ax)*rad_to_degree;
theta=atan2(sqrt((ax)^2+(ay)^2),az)*rad_to_degree;
psi=atan2(oz,-nz)*rad_to_degree;
```

2.Inverse Kinematic

輸入轉換矩陣後，將 kinematic table 的變數設定，並且取得 n、o、a 和 p 的值，開始計算各軸間的角度。

```
% input n o a p transformation matrix
T=input('\nT=[nx ox ax px ;  ny oy ay py ; nz oz az pz ;0 0 0 1;]\n');

%kinematic table parameter
a2 = 0.432;
a3 = -0.02;
d3 = 0.149;
d4 = 0.433;

%get n o a p value
nx = T(1,1);
ny = T(2,1);
nz = T(3,1);

ox = T(1,2);
oy = T(2,2);
oz = T(3,2);

ax = T(1,3);
ay = T(2,3);
az = T(3,3);

px = T(1,4);
py = T(2,4);
pz = T(3,4);
```

首先，先計算θ_1，依據公式(5)的結果可產生出兩組解，並將結果儲存至 array 中。

```
%calculate theta1
theta1= [atan2(py,px)-atan2(d3,sqrt(px^2+py^2-d3^2)) atan2(py,px)-atan2(d3,-1*sqrt(px^2+py^2-d3^2));];
```

$$\theta_1 = tan^{-1}\frac{p_y}{p_x} - tan^{-1}\frac{d_3}{\pm\sqrt{p_y^2 + p_x^2 - d_3^2}} \quad (5)$$

接著計算θ_3，依據公式(6)和(7)的結果也得出兩組解，將計算完的結果儲存至 array 中。

$$M = \frac{p_y^2 + p_x^2 + p_z^2 - a_2^2 - a_3^2 - d_3^2 - d_4^2}{2a_2} \quad (6)$$

$$\theta_3 = -tan^{-1}\frac{a_3}{d_4} + tan^{-1}\frac{M}{\pm\sqrt{d_4^2 + a_3^2 - M^2}} \quad (7)$$

```
M =((px)^2+(py)^2+(pz)^2-(a2)^2-(a3)^2-(d3)^2-(d4)^2)/(2*(a2));

%calculate theta3
theta3 = [-atan2(a3,d4)+atan2(M,sqrt(d4^2+a3^2-M^2)) -atan2(a3,d4)+atan2(M,-1*sqrt(d4^2+a3^2-M^2));];
```

```
%calculate theta2 4 5 6
for i = 1:2
    for j = 1:2

        %calculate theta2
        c1 = cos(theta1(j));
        s1 = sin(theta1(j));
        c3 = cos(theta3(i));
        s3 = sin(theta3(i));
        v1 = c1*px+s1*py;
        v2 = a3+a2*c3;
        v3 = d4+a2*s3;
        s23_t = v1*v3-v2*pz;
        c23_t = v1*v2+v3*pz;
        t23 = atan2(s23_t,c23_t);
        theta2 = t23 - theta3(i);
```

得到θ_1和θ_3後，開始計算θ_2，由於θ_1和θ_3各有兩組解，因此利用迴圈的方式來進行運算，並根據公式(8)和(9)解聯立得出θ_{23}，再減去θ_3，即可求得。

$$C_1C_{23}p_x + S_1C_{23}p_y - S_{23}p_z = a_3 + a_2C_3 \quad (8)$$

$$C_1S_{23}p_x + S_1S_{23}p_y + C_{23}p_z = d_4 + a_2S_3 \quad (9)$$

為了方便運算，將

$$C_1p_x + S_1p_y = v_1$$

$$a_3 + a_2C_3 = v_2$$

$$d_4 + a_2S_3 = v_3$$

便可得到

$$({v_1}^2 + {v_2}^2)S_{23} = v_1 * v_3 - v_2 * p_z$$

$$({v_1}^2 + {v_2}^2)C_{23} = v_1 * v_2 + v_3 * p_z$$

計算θ_{23}

$$\theta_{23} = tan^{-1}\frac{({v_1}^2 + {v_2}^2)S_{23}}{({v_1}^2 + {v_2}^2)C_{23}}$$

得出θ_2

$$\theta_2 = \theta_{23} - \theta_3$$

計算剩餘的θ_4、θ_5、θ_6，可根據下列式子的結果去計算出來，因為考慮到翻轉的問題，所以θ_4、θ_5、θ_6又會有另外一組解。

$$T_3^{-1} \cdot T_6 = {}^3T_6 = A_4 \cdot A_5 \cdot A_6$$

$$\begin{pmatrix} C_1C_{23} & S_1C_{23} & -S_{23} & -a_3 - a_2C_3 \\ -S_1 & C_1 & 0 & -d_3 \\ C_1S_{23} & S_1S_{23} & C_{23} & -a_2S_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^3T_6 = \begin{pmatrix} C_4C_5C_6 - S_4S_6 & -C_4C_5S_6 - S_4C_6 & C_4S_5 & 0 \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & S_4S_5 & 0 \\ -S_5C_6 & S_5S_6 & C_5 & d_4 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
%calculate theta4,5,6
c23 = cos(t23);
s23 = sin(t23);
c4s5 = c1*c23*ax + s1*c23*ay - s23*az;
s4s5 = -s1*ax + c1*ay;
c5 = c1*s23*ax + s1*s23*ay + c23*az;
s5c6 = -1*(c1*s23*nx+s1*s23*ny+c23*nz);
s5s6 = (c1*s23*ox+s1*s23*oy+c23*oz);

% case 1 theta 4 5 6
theta4 = atan2(s4s5, c4s5);
theta5 = acos(c5);
theta6 = atan2(s5s6, s5c6);

r2a(theta1(j),theta2,theta3(i),theta4,theta5,theta6);

% case 2 theta 4 5 6
theta4 = atan2(-s4s5, -c4s5);
theta5 = -acos(c5);
theta6 = atan2(-s5s6, -s5c6);

r2a(theta1(j),theta2,theta3(i),theta4,theta5,theta6);
```

最後利用 r2a 的函示將結果進行輸出。

- 加分題：討論兩種逆向運動學(代數法，幾何法)的優缺點

  逆向運動學:已知物體位置反推各個關軸的角度。

1. 幾何法

   甲、可以使用幾合關係和定理即可求解，對於空間概念有直接的關係。

   乙、在使用幾合法解決逆向運動學的問題時，有時只要獲得一組解就可以通過其對稱性得出其他的解。

2. 代數法

   甲、運算量極為複雜，有時會容易搞錯。

   乙、運用極座標的概念就可以往回推出各個軸之間的解。