```
%%File: CV-CODE-Project-R2PT-ANIMATION-3link-Quadruped-EXP-1.mlx    :      Robot Arm (2-link) T
% Trajectory Generation of the Equivalent End Effector _EE for a 2-link robot arm
% Source location: QRIS> C:\Users\USER-PC\QRIS\MATLAB Code
% ---------------------------------------------------------------------------------
% Notes:
% 1. Animation of original 3-link leg quadruped
% 2. Analysis of the configuration changes of the leg that is in contact with the ground level
% 3. Trajectories of J0 & J1 as generated by the front & back legs respectively
% 4. Item (3) above contributes to the forward locomotion of the quadruped
% --------------------------------------------------------------------------------------------
% --------------------------------------------------------------------------------------------
clc;clf ; clear all;
clear global
% ---------------------------------------------------------------------------
% CODE execution START:
% ---------------------------------------------------------------------------
global frI Ti Nnum nConfigs nDConfigs % declare all the global variables
nConfigs = 30;
nDConfigs = nConfigs-1;
% ---------------------------------------------------------------------------
% C:\Users\USER-PC\QRIS\MATLAB Code\DATA   <---- DATA Folder to access
theta=[];
MatrixDATA = dlmread('C:\Users\USER-PC\QRIS\MATLAB Code\DATA\PythonDATA-THETAdataLSQ-3.txt'); %
theta = MatrixDATA;% LSQ-optimised data [radians] <---> [300 x 8] <-- full Quadruped
[rws,cms] = size(MatrixDATA) ;
% ------------------------------------
% NOTE: angle order is [th6 th7 th8 th9 th1 th2 th3 th4]
% angle number          1    2   3   4   5   6   7   8
thetaSFLsys = theta(:,4:8); % <--- SFL system: [th9 th1 th2 th3 th4]
% angle number                                 4   5   6   7   8
thetaBLsys = theta(:,1:3); % <--- BL system: [th6 th7 th8]
% angle number                               1   2   3
% --------------------------------------------------------------------------------------------
% INTERPOLATE angle data so that 300 points are available for INVERSE KINEMATICS:::::::::
% --------------------------------------------------------------------------------------------
fr=30;T=1/fr;
x1 = linspace(0,(nConfigs-1)*T,nConfigs);% theta time line for all configurations
xx0 = linspace(x1(1),x1(end),10*nConfigs);% interpolated data x time line for angle
thetaINTsflSYS = zeros(300,5);
Y1=[];YY1=[];
for k=1:5
Y1 = thetaSFLsys(:,k)';
% INTERPOLATION ------------->
YY1 = interp1(x1,Y1,xx0,'spline','extrap');
thetaINTsflSYS(:,k) = YY1;
clearvars Y1 YY1
end
thetaSFL = thetaINTsflSYS; % LSQ-optimised data in RADIANS  [300 x 5]
% ----------------------------
% BL System Interpolated angles:
clearvars Y1 YY1
thetaINTblSYS = zeros(300,3);
```

```
Y1=[];YY1=[];
for k=1:3
Y1 = thetaBLsys(:,k)';
% INTERPOLATION ------------->
YY1 = interp1(x1,Y1,xx0,'spline','extrap');
thetaINTblSYS(:,k) = YY1;
clearvars Y1 YY1
end
thetaBL = thetaINTblSYS; % LSQ-optimised data in RADIANS  [300 x 3]
% ----------------------------------------------------------------------------------
MatrixDATA = [];
% -----------------------------------------------------------------------
% C:\Users\USER-PC\QRIS\MATLAB Code\DATA   <---- DATA Folder to access
phi=[];
MatrixDATA = dlmread('C:\Users\USER-PC\QRIS\MATLAB Code\DATA\PythonDATA-PHIdataLSQ-3.txt'); % p
phi = MatrixDATA;% LSQ-optimised data [radians] <---> [300 x 8] <-- full Quadruped
[rws,cms] = size(MatrixDATA) ;
% ------------------------------------
% NOTE: angle order is [ph6 ph7 ph8 ph9 ph1 ph2 ph3 ph4]
% angle number           1    2   3   4   5   6   7   8
phiSFLsys = phi(:,4:8); % <--- SFL system: [ph9 ph1 ph2 ph3 ph4]
% angle number                              4   5   6   7   8
phiBLsys = phi(:,1:3); % <--- BL system: [ph6 ph7 ph8]
% angle number                            1   2   3
% ----------------------------------------------------------------------------------
% INTERPOLATE angle data so that 300 points are available for INVERSE KINEMATICS::::::::
% ----------------------------------------------------------------------------------
fr=30;T=1/fr;
x1 = linspace(0,(nConfigs-1)*T,nConfigs);% phi time line for all configurations
xx0 = linspace(x1(1),x1(end),10*nConfigs);% interpolated data x time line for angle
phiINTsflSYS = zeros(300,5);
Y1=[];YY1=[];
for k=1:5
Y1 = phiSFLsys(:,k)';
% INTERPOLATION ------------->
YY1 = interp1(x1,Y1,xx0,'spline','extrap');
phiINTsflSYS(:,k) = YY1;
clearvars Y1 YY1
end
phiSFL = phiINTsflSYS; % LSQ-optimised data in RADIANS  [300 x 5]
% -----------------------------
% BL System Interpolated angles:
clearvars Y1 YY1
phiINTblSYS = zeros(300,3);
Y1=[];YY1=[];
for k=1:3
Y1 = phiBLsys(:,k)';
% INTERPOLATION ------------->
YY1 = interp1(x1,Y1,xx0,'spline','extrap');
phiINTblSYS(:,k) = YY1;
clearvars Y1 YY1
end
phiBL = phiINTblSYS; % LSQ-optimised data in RADIANS  [300 x 3]
```

```
% ------------------------------------------------------------------------------------------
% ------------------------------------------------------------------------------------------
MatrixDATA = [];
% ---------------------------------------------------------------------------
% C:\Users\USER-PC\QRIS\MATLAB Code\DATA   <---- DATA Folder to access
thetaD=[];
MatrixDATA = dlmread('C:\Users\USER-PC\QRIS\MATLAB Code\DATA\PythonDATA-thetaDmtx.txt'); % thet
thetaD =MatrixDATA; % [29x8]radians/s-->Python data
[rws,cms] = size(MatrixDATA) ;
% ------------------------------------
% NOTE: angle order is [th6 th7 th8 th9 th1 th2 th3 th4]
% angle number          1    2   3   4   5   6   7   8
thetaDSFLsys = thetaD(:,4:8); % <--- SFL system: [thD9 thD1 thD2 thD3 thD4] [rad/s]
% angle number                                     4    5    6     7   8
% ------------------------------------
% color CODE vectors ---->
cW = [1 1 1]; % white
cA = [1 0 0]; % red
cB = [0 0 1]; % blue
cC = [0 1 1]; % cyan
cD = [1 0 1]; % magenta
cE = [1 1 0]; % yellow
cJ = [0 1 0]; % green
cK = [0 0 0]; % black
% --- hybrids ----------
cF = [0.75 0 0.99]; % purple
cG = [0 0.4 0.3]; % dark green
cH = [0.6 0.98 0]; % light green
cI = [0.99 0.5 0]; % orange
% ------------------------------------------------------------------------------------------
% INTERPOLATE angle data so that 300 points are available for INVERSE KINEMATICS::::::::
% ------------------------------------------------------------------------------------------
fr=30;T=1/fr;
x1 = linspace(0,(nDConfigs-1)*T,nDConfigs);% theta time line for all configurations
xx0 = linspace(x1(1),x1(end),300);% interpolated data x time line for angle
Y1=[];YY1=[];
Y1 = thetaDSFLsys(:,5)'; % angular rates: End Effector = EE
% INTERPOLATION ------------->
YY1 = interp1(x1,Y1,xx0,'spline','extrap');
thetadEE=YY1;
% ------------------------------------------------------------------------------------------
thEE = thetaSFL(:,5);%[300pts] <--- EE
thdEE = thetadEE;
thdEEpython=thdEE;
% -----------------------------------------------------------------
Y1=[];YY1=[];thetad=[];
Y1 = thetaDSFLsys(:,4)'; % angular rates: thetaD3
% INTERPOLATION ------------->
YY1 = interp1(x1,Y1,xx0,'spline','extrap');
thetad=YY1;
% -----------------------------------------------------------------
th3INT = thetaSFL(:,4);%[300pts] <--- theta3
thd3INT = thetad;
```

```
thd3python=thd3INT;
% ------------------------------------------------------------------
% ------------------------------------------------------------------
Y1=[];YY1=[];thetad=[];
Y1 = thetaDSFLsys(:,3)'; % angular rates: thetaD2
% INTERPOLATION ------------->
YY1 = interp1(x1,Y1,xx0,'spline','extrap');
thetad=YY1;
% ------------------------------------------------------------------
th2INT = thetaSFL(:,3);%[300pts] <--- theta2
thd2INT = thetad;
thd2python=thd2INT;
% ------------------------------------------------------------------
% ------------------------------------------------------------------
% ------------------------------------------------------------------------------------
% ------------------------------------------------------------------------------------
% -------------------------------------
% Original LSQ link lengths:
% --------------------------
%  refJ0=o|--r5--|J9|--r1--|J1|--r2--|J2|--r3--|J3|--r4-->|J4=EE|
rlinkL = 0.2030; % <--- mean base length
r5 = (2.045*rlinkL) ;
r1 = (1.545*rlinkL) ;
r2 = (1.273*rlinkL) ;
r3 = (1.364*rlinkL) ;
r4 = rlinkL ; % <--- reference link length
r6 = (1.364*rlinkL) ;
r7 = (1.545*rlinkL) ;
r8 = (1.133*rlinkL) ;
% --------------------------
% thetaSFL (INTerpolated <--- 300 angle data points) data in [RADIANS] ----->
% SFL angles [rad]--->
th9 = thetaSFL(:,1);
th1 = thetaSFL(:,2);
th2 = thetaSFL(:,3);
th3 = thetaSFL(:,4);
th4 = thetaSFL(:,5);
DLSthEE = th4';
% BL angles [rad]--->
th6 = thetaBL(:,1);
th7 = thetaBL(:,2);
th8 = thetaBL(:,3);
% SFL phi angles [rad]--->
ph9 = phiSFL(:,1);
ph1 = phiSFL(:,2);
ph2 = phiSFL(:,3);
ph3 = phiSFL(:,4);
ph4 = phiSFL(:,5);
DLSphEE = ph4';
% BL angles [rad]--->
ph6 = phiBL(:,1);
ph7 = phiBL(:,2);
ph8 = phiBL(:,3);
```

```
% --------------
Nnum = rws; % number of angle samples
indx=1:1:Nnum;
%fr=30; % frame rate of original video sequence
%T=1/fr; % time interval between configurations (original video)
frI = 300; % frame rate for interpolated data
Ti = 1/frI; % time interval between configurations for interpolated data
x0LSQ=0;y0LSQ=0;z0LSQ=0; % origin of SFL system = COM(x,y,z)
% ---------------------------------------------------------
% FL EE coordinates 3D --------------------------->
% ---------------------------------------------------------
xEEspace = x0LSQ + r5*cos(th9).*cos(ph9) + r1*cos(th9 + th1).*cos(ph9 + ph1) + r2*cos(th9 + th1
yEEspace = y0LSQ + r5*sin(th9) + r1*sin(th9 + th1) + r2*sin(th9 + th1 + th2) + r3*sin(th9 + th1
zEEspace = z0LSQ + r5*cos(th9).*sin(ph9) + r1*cos(th9 + th1).*sin(ph9 + ph1) + r2*cos(th9 + th1
% ---------------------------------------------------------
% ---------------------------------------------------------
% BL EE coordinates 3D --------------------------->
% ---------------------------------------------------------
xEEspaceBL = x0LSQ + r6*cos(th6).*cos(ph6) + r7*cos(th6 + th7).*cos(ph6 + ph7) + r8*cos(th6 + t
yEEspaceBL = y0LSQ + r6*sin(th6) + r7*sin(th6 + th7) + r8*sin(th6 + th7 + th8);
zEEspaceBL = z0LSQ + r6*cos(th6).*sin(ph6) + r7*cos(th6 + th7).*sin(ph6 + ph7) + r8*cos(th6 + t
% ------------------------------------------------
% ALL JOINT coordinates of 3-link legged quadruped:
% ------------------------------------------------
x0LSQ=0;y0LSQ=0;z0LSQ=0; % origin of system = COM(x,y,z) = J0
p0x = x0LSQ*ones(300,1); p0y = y0LSQ*ones(300,1);p0z = z0LSQ*ones(300,1); % origin of system =
p9x = x0LSQ + r5*cos(th9).*cos(ph9); p9y = y0LSQ + r5*sin(th9);p9z = z0LSQ + r5*cos(th9).*sin(p
p1x = x0LSQ + r5*cos(th9).*cos(ph9) + r1*cos(th9 + th1).*cos(ph9 + ph1); p1y = y0LSQ + r5*sin(t
p2x = x0LSQ + r5*cos(th9).*cos(ph9) + r1*cos(th9 + th1).*cos(ph9 + ph1) + r2*cos(th9 + th1 + th
p3x =  x0LSQ + r5*cos(th9).*cos(ph9) + r1*cos(th9 + th1).*cos(ph9 + ph1) + r2*cos(th9 + th1 + t
p4x = xEEspace;p4y = yEEspace;p4z = zEEspace;
% --------
p6x =  x0LSQ + r6*cos(th6).*cos(ph6);p6y = y0LSQ + r6*sin(th6);p6z =  z0LSQ + r6*cos(th6).*sin(
p7x =  x0LSQ + r6*cos(th6).*cos(ph6) + r7*cos(th6 + th7).*cos(ph6 + ph7) ;p7y =  y0LSQ + r6*sin
p8x = xEEspaceBL;p8y = yEEspaceBL;p8z = zEEspaceBL;

xPTS_SFL = [p0x,p9x,p1x,p2x,p3x,p4x]; % [300x6] <--- configuration coordinates : X
yPTS_SFL = -1.*[p0y,p9y,p1y,p2y,p3y,p4y]; % [300x6] <--- configuration coordinates : Y
xPTS_BL = [p0x,p6x,p7x,p8x]; % [300x3] <--- configuration coordinates : X
yPTS_BL = -1.*[p0y,p6y,p7y,p8y]; % [300x3] <--- configuration coordinates : Y
% ----------------------------
GNDcontactBL = min(-p8y(113:201));
GNDcontactBL
```

```
GNDcontactBL = -0.7118
```

```
INDX_gnd = find(-p8y==GNDcontactBL); % <--- ground contact BL index for gait 2
INDX_gnd
```

```
INDX_gnd = 142
```

```
%GNDcntctBL_pnt = [xEE_BLspace(INDX_gnd); yEE_BLspace(INDX_gnd)]; % ground contact point [x;y]
%GNDcntctBL_pnt
```

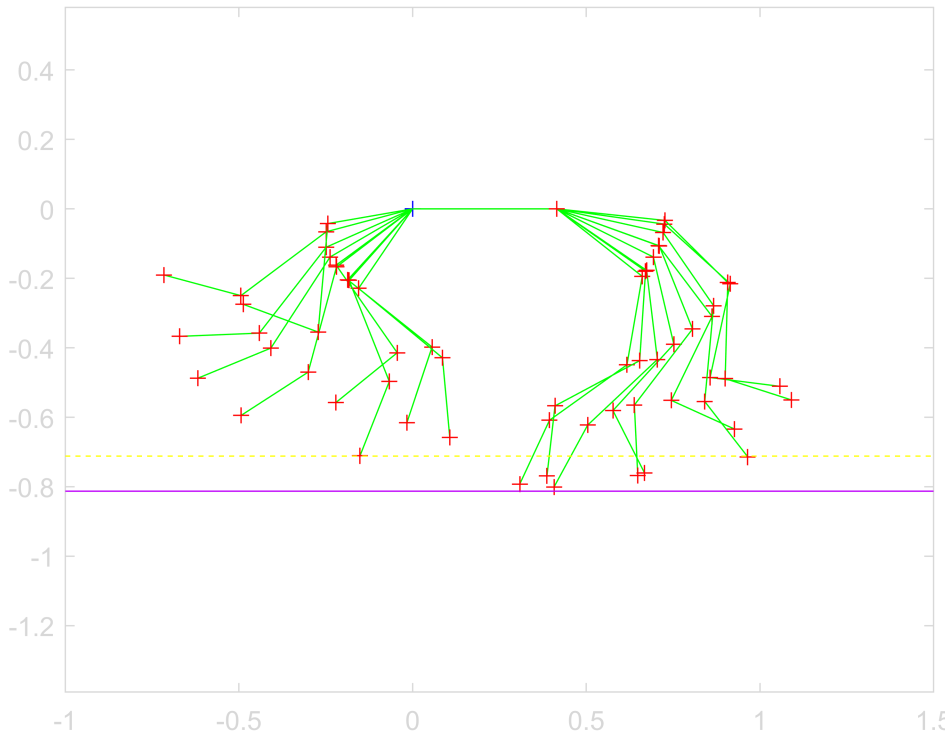
```
GNDLL = [[-1.0,-0.9,-0.5,0,0.5,1,1.25,1.35,1.5];GNDcontactBL*ones(1,9)];

GNDcontactFL = min(-p4y(113:201));
GNDcontactFL
```

```
GNDcontactFL = -0.8128
```

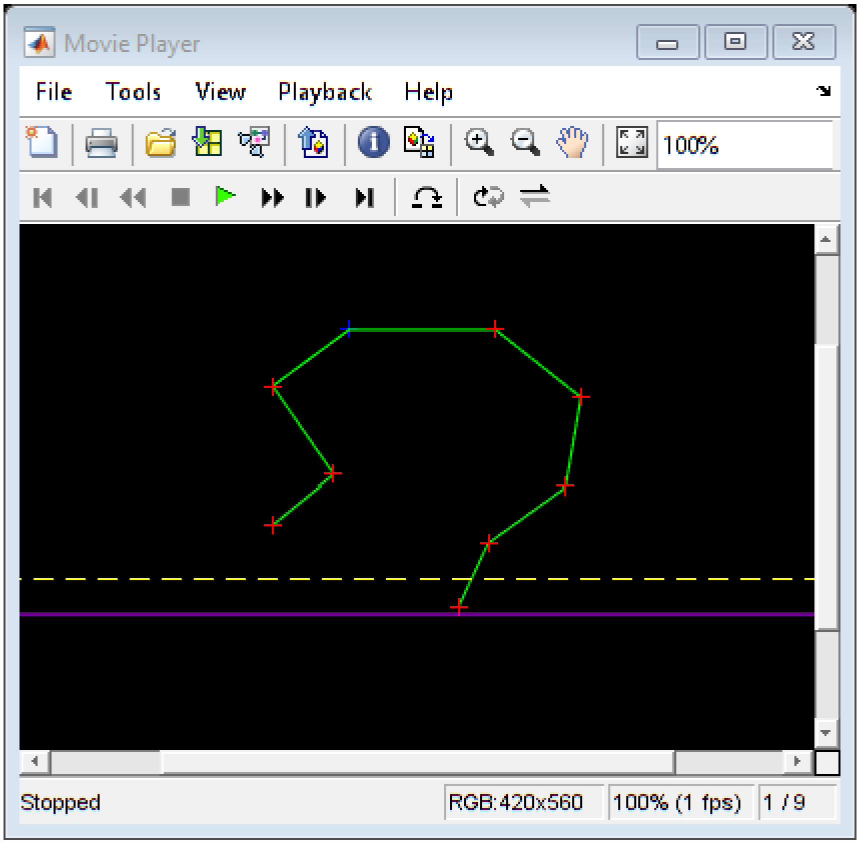
```
INDX_gnd_FL = find(-p4y==GNDcontactFL); % <--- ground contact FL index for gait 2
INDX_gnd_FL
```

```
INDX_gnd_FL = 197
```

```
GNDLL_FL = [[-1.0,-0.9,-0.5,0,0.5,1,1.25,1.35,1.5];GNDcontactFL*ones(1,9)];

loops = 1;
movColr(1:loops) = struct('cdata', [],'colormap',[]);
i=1;
% -----------------------------
figure
% -----------------------------
% set background colour
fig = gcf;
fig.Color = [0 0 0]; % black = [0 0 0]
colordef black
k=0;

while k < 90

plot(p0x,p0y,'b','Marker','+') % reference marker = CoM LSQ
hold on
line(xPTS_SFL(113+k,:),yPTS_SFL(113+k,:), 'Color',cJ,'LineStyle','-') ; % <---- dynamic links
line(xPTS_BL(113+k,:),yPTS_BL(113+k,:), 'Color',cJ,'LineStyle','-') ; % <---- dynamic links

plot(xPTS_SFL(113+k,2),yPTS_SFL(113+k,2),'r','Marker','+') % <---- all joint 9 markers
plot(xPTS_SFL(113+k,3),yPTS_SFL(113+k,3),'r','Marker','+') % <---- all joint 1 markers
plot(xPTS_SFL(113+k,4),yPTS_SFL(113+k,4),'r','Marker','+') % <---- all joint 2 markers
plot(xPTS_SFL(113+k,5),yPTS_SFL(113+k,5),'r','Marker','+') % <---- all joint 3 markers
plot(xPTS_SFL(113+k,6),yPTS_SFL(113+k,6),'r','Marker','+') % <---- all joint 4 markers
% --------
plot(xPTS_BL(113+k,2),yPTS_BL(113+k,2),'r','Marker','+') % <---- all joint 6 markers
plot(xPTS_BL(113+k,3),yPTS_BL(113+k,3),'r','Marker','+') % <---- all joint 7 markers
plot(xPTS_BL(113+k,4),yPTS_BL(113+k,4),'r','Marker','+') % <---- all joint 8 markers
% --------
plot(GNDLL_FL(1,:),GNDLL_FL(2,:), 'Color',cF,'LineStyle','-'); % GROUND level for BL
plot(GNDLL(1,:),GNDLL(2,:), 'Color',cE,'LineStyle','--'); % GROUND level for FL

axis equal  % <----------------- SET axes equal for plot
k=k+10;
movColr(i) = getframe(gcf); % <--- store the current frame
i = i + 1;
end
hold off
```




```
%mplay(movColr);  % default : 100% <---> 20 frames/second = 20fps
%fps = 1/3; % <--- shows a frame every 1/3 = 0.333 [s]
fps = 1; % <--- shows a frame every 1 [s]
implay(movColr,fps); % specify the frames per second to show in the animation
```




```
clear movColr
% ----------------------------------------------------------------------------------
```