

```

% Pl Q(c). A function that takes contact points and desired body configs as the input and output the required joint angles for four legs

function [joint_angles,message] = getJointAngles(body_configs,body_consts,contact_pts)
addpath('/Users/guoyongxin/Desktop/Assignment_Academics/Senior_Second semester/MEC529/Myfunctions');
% contact_pts is a 4 by 4 matrix that consists of foot contact points
% for each of four legs using homogenous representation.
% body_configs is a transformation matrix
% body_consts stores constants associated with the robot, such as
% link length, body length and width. body_consts=[l1;l2;w;l];
w = body_consts(3); % x-component distance from the hip to the body center.
l = body_consts(4); % y-component distance from the hip to the body center.
l1 = body_consts(1); % link length 1
l2 = body_consts(2); % link length 2
I = eye(3);
% assign joint angles beforehand. it is 4 by 4 matrix. Each 2 rows has
% one set of soln, there are 2 set of solns in total for each contact pts.
joint_angles = zeros(4,4);
legNum = 4; % assign leg number.
T_BToW = body_configs; % assign body config to variable T_BtoW for clearance.
T_BToW_inv = getTransInv(T_BToW); % get inverse transformation matrix
% assign the position vector between the body frame to the hip frames
% expressed in body frame.
P_bToh1 = [w;-l;0];
P_bToh2 = [-w;-l;0];
P_bToh3 = [-w;l;0];
P_bToh4 = [w;l;0];
% put them in a aggregated matrix for being easier to assign them in
% the transformation matrix from hip frame to body frame.
P_bTohi = [P_bToh1,P_bToh2,P_bToh3,P_bToh4];
counter = 0;
message = "The leg with inaccessible points are: ";
for i = 1:legNum
    T_HiToB = [I,P_bTohi(:,i);[0 0 0],1]; % transformation matrix from hip to body
    T_HiToB_inv = getTransInv(T_HiToB); % get inverse of T_HiToB.
    P_tiToW = contact_pts(:,i); % get position vector of contact pts expressed in world frame {W}
    P_tiToHi = T_HiToB_inv * T_BToW_inv * P_tiToW; % get contact pts expressed in hip frame {H} for each leg
    y = P_tiToHi(2); % get y-component of the contact pts in {H};
    z = P_tiToHi(3); % get z-component of the contact pts in {H};
    % input the y and z value of the point into RR_inverse_2D function.
    theta_i = RR_inverse_2D([l1;l2],[y;z]); % get inverse solns
    if isstring(theta_i) == 1 % when it is a string,it means error occurs.
        joint_angles(:,i) = [NaN;NaN;NaN;NaN]; % means inaccessible point
        counter = counter + 1; % update inaccessible point counter.
        message = message + num2str(i) + " ";
    else % if it is not a string, assign theta.
        joint_angles(1:2,i) = theta_i(:,1); % assign first set of soln for each leg.
        joint_angles(3:4,i) = theta_i(:,2); % assign second set of soln for each leg.
    end
end

if counter == 0 % if all the points are accessible.
    message = "All the points are accessible";
else
end

end
end

```

Not enough input arguments.

Error in getJointAngles (line 10)
w = body_consts(3); % x-component distance from the hip to the body center.

