

Contents

- create Tse positions for all 8 trajectories
- add the state of the gripper for each trajectory
- simulate Trajectorys between all 8 Tse positions
- Output array to config_array.csv

```
function [outputConfigs] = TrajectoryGenerator(TseI,TscI, TscF, TceGrasp, TceStandoff, k)
```

```
% Trajectory Generator
% Takes TseI: The initial config of the EE
%     TscI: The initial config of the cube
%     TscF: The final config of the cube
%     TceGrasp: The config of the EE relative to the cube
%     TceStandoff: The config of the EE relative to the cube to approach
%     the cube from
%     k: number of trajectory reference configs per 0.01 seconds. must be
%     1 or greater
% Returns outoutConfigs: an array containing all the configurations of the
%     EE, one T per line, in order: r11,r12,r13,r21,r22,r23,r31,r32,r33,
%     px, py, pz, gripperstate

% This function calculates a trajectory corresponding to the screw motion
% from TseI through all 8 defined trajectories. Outputs a CSV file that can
% be loaded into CoppeliaSim to simulate EE trajectory for the task
if k<1
    error('k not greater then 1')
end
```

Not enough input arguments.

Error in TrajectoryGenerator (line 18)
if k<1

create Tse positions for all 8 trajectories

```
Startpos = TseI;
pos{1} = TscI*TceStandoff;    %position of EE at block location, adjusted by standoff matrix
pos{2}= TscI*TceGrasp;        % position of EE at block pickup location and oreintation
pos{3}= pos{2};               %same postion, just close the gripper and wait
pos{4}= pos{1};               %back up to Pickup Standoff position, just with gripper closed
pos{5}= TscF*TceStandoff;
pos{6}= TscF*TceGrasp;
pos{7}= pos{6};               %same position just open gripper and wait
pos{8}= pos{5};               %back up to Final Standoff Position, with the gripper open
```

add the state of the gripper for each trajectory

```
Gripperstate = [0, 0, 1, 1, 1, 1, 0, 0, 0];
```

simulate Trajectorys between all 8 Tse positions

```
%function is traj = ScrewTrajectory(Xstart, Xend, Tf, N, method)
% set the time in seconds for each motion to occur
t= [5 1 1 1 4 1 1 1];
N= t.*k/.01; %how many intermeadeate steps are needed for the motion to last T seconds
traj(1,:)= ScrewTrajectory(Startpos, pos{1}, t(1), N(1),5);
for n=2:8
    traj(n,1:N(n))=ScrewTrajectory(pos{n-1}, pos{n}, t(n), N(n),5);
end
```

Output array to config_array.csv

config_array.csv will be located in Matlab's current directory.

```
% preallocate size of our output matrix
leng= sum(N);
outputConfigs = zeros(leng,13);
line = 1; %a counter to output each Transformation matrix as a line
for n=1:8 %n cycles through the 8 traj
    for i= 1:N(n) % i cycles through the T matrices inside each traj
        outputConfigs(line,:) = [traj{n,i}(1,1),traj{n,i}(1,2),traj{n,i}(1,3),traj{n,i}(2,1),traj{n,i}(2,2),...
            traj{n,i}(2,3),traj{n,i}(3,1), traj{n,i}(3,2),traj{n,i}(3,3),traj{n,i}(1,4),traj{n,i}(2,4),traj{n,i}(3,4),Gripperstate(n)];
        line =line+1;
    end
end
writematrix(outputConfigs,'config_array.csv')
```

```
end
```