
Simulation

```
fprintf('\n');
clearvars -except function_list pub_opt
close all
theta0_121=[45;60;-75]*pi/180;%rad

fprintf('MRP vector:\n');
MRP = DCM2MRP(Euler2DCM('121',theta0_121));
printVector(MRP, '');
omega_body = [1; -2; 0.5]*pi/180; %rad/s
I_body=[14 0 0;0 16 0;0 0 19]; %kg*m2
cm_torque=[0;0;0];

delta_t = 0.01;
t_end = 600 - delta_t; % seconds

% Arrays for recording and plotting
t_mat = 0:delta_t:t_end+delta_t;
[rows, cols] = size(t_mat);
MRP_mat = zeros(3,cols);
omega_mat = zeros(3,cols);
MRP_mat(:,1) = MRP;
omega_mat(:,1) = omega_body;
idx = 2;

% RK4 integration
state = [MRP; omega_body];
for t = 0:delta_t:t_end
    k1 = [derivMRP(state(1:3), state(4:6)); ...
          dBodyRatesRigid(state(4:6), I_body, cm_torque)];
    k2 = [derivMRP(state(1:3) + delta_t*k1(1:3)/2, ...
          state(4:6) + delta_t*k1(4:6)/2); ...
          dBodyRatesRigid(state(4:6) + delta_t*k1(4:6)/2, I_body, cm_torque)];
    k3 = [derivMRP(state(1:3) + delta_t*k2(1:3)/2, ...
          state(4:6) + delta_t*k2(4:6)/2); ...
          dBodyRatesRigid(state(4:6) + delta_t*k2(4:6)/2, I_body, cm_torque)];
    k4 = [derivMRP(state(1:3) + delta_t*k3(1:3), ...
          state(4:6) + delta_t*k3(4:6)); ...
          dBodyRatesRigid(state(4:6) + delta_t*k3(4:6), I_body, cm_torque)];

    state = state + delta_t/6*(k1 + 2*k2 + 2*k3 + k4);

    % Enforce |MRP| <= 1, switch to shadow set if needed
    if norm(state(1:3)) > 1
        state(1:3) = -state(1:3)/dot(state(1:3), state(1:3));
    end

    % Updating array
    MRP_mat(:,idx) = state(1:3);
    omega_mat(:,idx) = state(4:6);
    idx = idx + 1;
end
```

```

end

font_size=8;
figure
plot(t_mat, MRP_mat);
% title('MRP Propagation','FontSize',font_size)
xlabel('time(s)','FontSize',font_size)
ylabel('Element Magnitude','FontSize',font_size)
legend('\sigma_{1}', '\sigma_{2}', '\sigma_{3}')
grid on
set(gca,'FontSize',font_size)
fprintf('\n\n\n');
figure
plot(t_mat, omega_mat*180/pi);
% title('Body Rates','FontSize',font_size)
xlabel('time(s)','FontSize',font_size)
ylabel('Element Magnitude (deg/s)','FontSize',font_size)
legend('\omega_{1}', '\omega_{2}', '\omega_{3}')
grid on
set(gca,'FontSize',font_size)
fprintf('\n\n\n');

simulated_H=zeros(1,cols);
simulated_T=zeros(1,cols);
H_init=norm(I_body*omega_mat(:,1));
T_init=0.5*dot(I_body*omega_mat(:,1),omega_mat(:,1));
for val_idx=1:cols
    simulated_H(val_idx)=norm(I_body*omega_mat(:,val_idx));
    simulated_H(val_idx)=abs(H_init-simulated_H(val_idx));
    simulated_T(val_idx)=0.5*dot(I_body*omega_mat(:,val_idx),omega_mat(:,val_idx));
    simulated_T(val_idx)=abs(T_init-simulated_T(val_idx));
end
figure
semilogy(t_mat, simulated_H);
% title('Angular Momentum Change','FontSize',font_size)
xlabel('time(s)','FontSize',font_size)
ylabel('Angular Momentum \Delta (N*m*s)','FontSize',font_size)
legend('\Delta H')
grid on
set(gca,'FontSize',font_size)
fprintf('\n\n\n');
figure
semilogy(t_mat, simulated_T);
xlabel('time(s)','FontSize',font_size)
ylabel('Rotational Kinetic Energy \Delta (J)','FontSize',font_size)
legend('\Delta T')
grid on
set(gcf,'FontSize',font_size)
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 10 5])
axis tight
print(gcf, '-dpng', 'blah')
% title('Rotational Kinetic Energy Change','FontSize',font_size)
fprintf('\n\n\n');

```

Published with MATLAB® R2013b