Table of Contents


```
function manifolds(tout, xout, mu, l1_pos, manifold_time)
    % Script to compute stable/unstable manifolds for a periodic orbit
    % Inputs:
    % tout - discrete time steps
    % xout - 42x1 discrete state vectors
    % mu - system mass ratio
    % l1_pos - equilibrium point position
    % manifold_time - time to propagate manifold forward/backward
    %
    % Outputs:
    % Graph with stable/unstable manifolds

    % Set options for ode113()
    % Part b
    % options = odeset('RelTol', 1e-12, 'AbsTol', 1e-12, 'Events', @(t,y)
eventFn(t, y, mu));

    % Part c - ignore event function
    options = odeset('RelTol', 1e-12, 'AbsTol', 1e-12);

    a = 384400; % [kg] EM average SMA
    d = 50 / a; % [-] Unitless, normalized by a

    period = tout(end);

    p1_pos = [-mu, 0, 0];
    p2_pos = [1-mu, 0, 0];

    figure()
    plot(xout(:,1), xout(:,2), 'black', 'LineWidth', 3)
    hold on
    scatter(l1_pos(1), l1_pos(2), 'filled', 'red')
    scatter(p1_pos(1), p1_pos(2), 'filled', 'blue')
    scatter(p2_pos(1), p2_pos(2), 'filled', ' black')

    % Compute STM - phi(t1+T, t1)
    phi_t1T_t1 = reshape(xout(end,7:42), [6,6])';

    % Begin for loop
    for i = 1:10:length(tout)

        % Compute STM - phi(tj+T, tj)
        phi_tj_t1 = reshape(xout(i, 7:42), [6,6])';
        phi_tjT_tj = phi_tj_t1 * phi_t1T_t1 * inv(phi_tj_t1);

        % Get evals, evecs
        [V, D] = eig(phi_tjT_tj);
```

```
% Get evals as an array
for j = 1:6
    evals(j) = D(j,j);
end

% Subtract evals by 1 and get 2 minimum indices. These are trivial
% indices
evals_minus_1 = evals - 1;
[min_evals, trivial_index] = mink(evals_minus_1, 2);

% If eval is real and not trivial, assign stable and unstable
% indices
for j = 1:6
    if (isreal(evals(j)) && isnotin(trivial_index, j))
        if evals(j) < 1
            stable_index = j;
        elseif evals(j) > 1
            unstable_index = j;
        end
    end
end

% Get stable/unstable evec and normalize eigenvector by 1st 3 terms
stable_eval = D(stable_index, stable_index);
stable_evec = V(:, stable_index);
stable_pos_norm = norm(stable_evec(1:3));
stable_evec = stable_evec/stable_pos_norm;
% stable_evec(4:6) = -stable_evec(4:6);
unstable_eval = D(unstable_index, unstable_index);
unstable_evec = V(:, unstable_index);
unstable_pos_norm = norm(unstable_evec(1:3));
unstable_evec = unstable_evec/unstable_pos_norm;

% Step into manifold
x_manifold_s_p = xout(i,1:6)' + d * stable_evec;
x_manifold_s_n = xout(i,1:6)' - d * stable_evec;

% ONLY FOR L1
% If x-velocity is positive, moon-bound
% If x-velocity if negative, earth-bound
if (x_manifold_s_p(4) > 0)
    moon_stable = x_manifold_s_p;
    earth_stable = x_manifold_s_n;
else
    moon_stable = x_manifold_s_n;
    earth_stable = x_manifold_s_p;
end
% Repeat for unstable manifolds
x_manifold_u_p = xout(i,1:6)' + d * unstable_evec;
x_manifold_u_n = xout(i,1:6)' - d * unstable_evec;
if (x_manifold_u_p(4) > 0)
    moon_unstable = x_manifold_u_p;
    earth_unstable = x_manifold_u_n;
```

```
        else
            moon_unstable = x_manifold_u_n;
            earth_unstable = x_manifold_u_p;
        end

        % Propagate using the event functions
        [moon_stable_t, moon_stable_x] = ode113(@(t, state)CR3BP(state, mu),
[0, -manifold_time], moon_stable, options);
        [moon_unstable_t, moon_unstable_x] = ode113(@(t, state)CR3BP(state,
mu), [0, manifold_time], moon_unstable, options);
        [earth_stable_t, earth_stable_x] = ode113(@(t, state)CR3BP(state,
mu), [0, -manifold_time], earth_stable, options);
        [earth_unstable_t, earth_unstable_x] = ode113(@(t, state)CR3BP(state,
mu), [0, manifold_time], earth_unstable, options);

        plot(moon_stable_x(:,1), moon_stable_x(:,2), 'blue')
        plot(earth_stable_x(:,1), earth_stable_x(:,2), 'blue')
        plot(moon_unstable_x(:,1), moon_unstable_x(:,2), 'red')
        plot(earth_unstable_x(:,1), earth_unstable_x(:,2), 'red')

    end
    hold off
    legend("Lyapunov Orbit", "L1", "Earth", "Moon")
    grid on
    axis equal
    xlabel('$$\hat{x}$$','Interpreter','Latex', 'FontSize',18)
    ylabel('$$\hat{y}$$','Interpreter','Latex', 'FontSize',18)
end

Not enough input arguments.

Error in manifolds (line 23)
    period = tout(end);
```

Published with MATLAB® R2024a