Make graphs

Trevor Burgoyne 13 Nov 2022

```
function [avg_hss, var_hss, avg_kt, var_kt, kphat, kdhat, kp, kd] = make_graphs(ROOT_DIR, PREFIX, DISP_NAME, LABEL_NAME, N_TESTS, N_RUNS, masses)
    COLORS = ["red", "blue", "green", "black"];
    T_START = 20; % s
    T_STEADY = 30; % s, Chosen as start of steady-state response from observation
    T_END = T_START+25; % s
    HREF = 1.25; % m
    g = 9.81; % m/s^2

    % Plotting options
    font_size = 12;
    line_size = 15;
    line_width = 1;
    avg_hss = zeros(1,N_TESTS);
    var_hss = zeros(1,N_TESTS);
    avg_kt  = zeros(1,N_TESTS);
    var_kt  = zeros(1,N_TESTS);
    kphat   = zeros(1,N_TESTS);
    kdhat   = zeros(1,N_TESTS);
    kp      = zeros(1,N_TESTS);
    kd      = zeros(1,N_TESTS);
    for test_n=1:N_TESTS
        ts = zeros(1,N_RUNS);
        kt = zeros(1,N_RUNS);
        hss = zeros(1,N_RUNS);
        zs = zeros(1,N_RUNS);
        figure(test_n)
        hold on
        for run_n=1:N_RUNS
            % Load data
            path = ROOT_DIR + PREFIX + test_n;
            if N_RUNS > 1 % Add runs suffix if more than one run was done
                path = path + "R" + run_n;
            end
            path = path + ".mat";
            load(path);

            if run_n == 1 % Only plot reference once
                plot(time,-z_ref,'Linewidth',line_width,'Color',COLORS(4),'DisplayName','z_{ref}');
            end
            plot(time,-z_est,'Linewidth',line_width,'Color',COLORS(run_n),'DisplayName',LABEL_NAME + " " + run_n);

            % Steady-state error
            idxs = find(time >= T_STEADY & time <=T_END); % Indices of steady state region
            z_arr = -z_est(idxs); % Z values being investigated
            zs(run_n) = double(mean(z_arr)); % m, Experimental settling value
            hss(run_n) = zs(run_n) - HREF; % m, Steady state error

            % Settling time
            idxs = find(time >= T_START & time <=T_END); % Idxs of investigated response
            start_idx = idxs(1);
            z_arr = -z_est(idxs); % Z values being investigated

            % Find last time z dipped below 95% of z_settle
            ts_idxs = find(z_arr <= 0.95*zs(run_n));
            if isempty(ts_idxs)
                ts1 = 0;
            else
                ts1 = time(ts_idxs(end) + start_idx);
            end

            % Find last time z rose above 105% of z_settle
            ts_idxs = find(z_arr >= 1.05*zs(run_n));
            if isempty(ts_idxs)
                ts2 = 0;
            else
                ts2 = time(ts_idxs(end) + start_idx);
            end

            ts(run_n) = max(ts1,ts2); % s, Settling time (use the later time)

            % Kt calculation
            motors = abs([motor1' motor2' motor3' motor4']); % Get all motor values
            u = mean(motors);
            kt(run_n) = (masses(test_n)*g) / (4*u) * 1000; % N

        end

        scale = [1.05 0.95]; % scalar values used to position text
        for run_n=1:N_RUNS
            % Steady-state error
            if zs(run_n) == max(zs) % Position labels above if the line is higher
                y = scale(1);
            else
                y = scale(2);
            end
            line_name = "h_{ss} = " + hss(run_n);
            text(T_STEADY,y*zs(run_n),line_name,'Color',COLORS(run_n))
            yline(zs(run_n),"--",'Linewidth',line_width,'Color',COLORS(run_n),'HandleVisibility','off')

            % Settling time
            if ts(run_n) == max(ts) % Position labels above if the line is higher
                y = scale(1);
            else
                y = scale(2);
            end
            line_name = "ts = " + (ts(run_n)-T_START);
            text(1.01*max(ts),y*.7,line_name,'Color',COLORS(run_n))
            xline(ts(run_n),"--",'Linewidth',line_width,'Color',COLORS(run_n),'HandleVisibility','off')
        end
        title(sprintf('%s: $\\hat{K_{p}} = %s, \\hat{K_{d}} = %s$',DISP_NAME,num2str(Kp),num2str(Kd)),'Interpreter','latex');
        xlabel('Time (s)','fontsize',font_size);
        ylabel('Altitude (m)','fontsize',font_size);
        legend('show','Location','best');
        set(gca,'XMinorGrid','off','GridLineStyle','-','FontSize',line_size)
        xlim([T_START-1 T_END+1]);
        ylim([0.4 1.5]);
        grid on

        avg_hss(test_n) = mean(hss);
        var_hss(test_n) = var(hss);
        avg_kt(test_n)  = mean(kt);
        var_kt(test_n)  = var(kt);
        kphat(test_n)   = Kp;
        kdhat(test_n)   = Kd;
        kp(test_n)      = Kp/(4*avg_kt(test_n));
        kd(test_n)      = Kd/(4*avg_kt(test_n));
    end
end
```

```
Not enough input arguments.

Error in make_graphs (line 16)
    avg_hss = zeros(1,N_TESTS);
```

Published with MATLAB® R2020b