

---

# Final Exam Problem 1

## Table of Contents

Initialize .....	1
Smoothing .....	2

## Initialize

```
clearvars -except function_list pub_opt P_joseph_store
global function_list;
function_list = {};
close all

obs_data = load('hw11.dat');

T = 10;
truth = sin(obs_data(:,1) * 2*pi/T);
main_plot = figure;
hold on
% plot(obs_data(:,1),obs_data(:,2))
plot(obs_data(:,1),truth,'LineWidth',3)

eta0_ap = 0;
P0_ap = 1;
R = 1;
Q = 1;
num_obs = length(obs_data(:,1));
eta_est_store = zeros(num_obs,1);

eta_est = eta0_ap;
P = P0_ap;

tc_vec = [1/0.045];
sig_vec = [2.49];

best_RMS = -1;
for idx = 1:length(tc_vec)
    time_const = tc_vec(idx);
    beta = 1/time_const;
    sigma = sig_vec(idx);
    eta_est_store_inner = zeros(num_obs,1);
    RMS_accum = 0;
    for ii = 1:num_obs
        % STM
        if ii == 1 %measurement at t = 0
            m = 1;
        else
            m = exp(-beta*(obs_data(ii,1)-obs_data(ii-1,1)));
```

```

end
STM = m;

% Time Update
eta_ap = STM*eta_est;
gamma = sqrt(sigma*sigma/2/beta*(1-m*m));
P_ap = STM*P*STM + gamma*Q*gamma;

% Kalman gain
K = P_ap/(P_ap+1); % valid for this 1D case

% Measurement Update
Y = obs_data(ii,2);
eta_est = eta_ap +K*(Y-eta_ap); %H~ == 1 in this case.
P = K;

eta_est_store_inner(ii) = eta_est;
RMS_accum = RMS_accum + (truth(ii)-eta_est)*(truth(ii)-eta_est);

% Stores
P_store(ii) = P;
STM_store(ii) = STM;
gamma_store(ii) = gamma;
end
RMS = sqrt(RMS_accum/num_obs);
fprintf(sprintf('RMS for tau=%f, sigma=%f: %f\n',time_const, sigma, RMS));
if best_RMS == -1
    best_RMS = RMS;
    eta_est_store = eta_est_store_inner;
else
    if RMS < best_RMS
        eta_est_store = eta_est_store_inner;
    end
    best_RMS = min(best_RMS,RMS);
end
end
plot(obs_data(:,1),eta_est_store,'r','LineWidth',2)

```

## Smoothing

```

smoothed_store = zeros(num_obs,1);
smoothed_store(end) = eta_est_store(end);
for ii = num_obs-1:-1:1
    P = P_store(ii);
    STM = STM_store(ii+1);
    gamma_store(ii+1);
    eta_est = eta_est_store(ii);
    eta_last = smoothed_store(ii+1);
    S = P*STM/(STM*P*STM + gamma*Q*gamma);
    eta_est_new = eta_est + S*(eta_last-STM*eta_est);
    smoothed_store(ii) = eta_est_new;
end
figure(main_plot);

```

```
plot(obs_data(:,1),smoothed_store,'k','LineWidth',3)
RMS_smooth = sqrt(sum((truth-smoothed_store).*(truth-smoothed_store))/...
    /num_obs);
legend('Truth', sprintf('Filtered Data, RMS = %f',best_RMS),...
    sprintf('Smoothed Filter, RMS = %f',RMS_smooth))
xlabel('Time'), ylabel('eta')
fprintf(sprintf('Smoothed RMS: %f\n',RMS_smooth));
fprintf('The histogram of the smoothed results is more Gaussian.\n')
figure
hist(obs_data(:,2)-eta_est_store,25);
title('Filter Residuals')
xlabel('Eta residual')
figure
hist(obs_data(:,2)-smoothed_store,25);
title('Smoothed Filter Residuals')
xlabel('Eta residual')
```

*Published with MATLAB® R2013b*