Exercise Session 2: Plasma current and position determination

Table of Contents

Load the data and get plasma grid

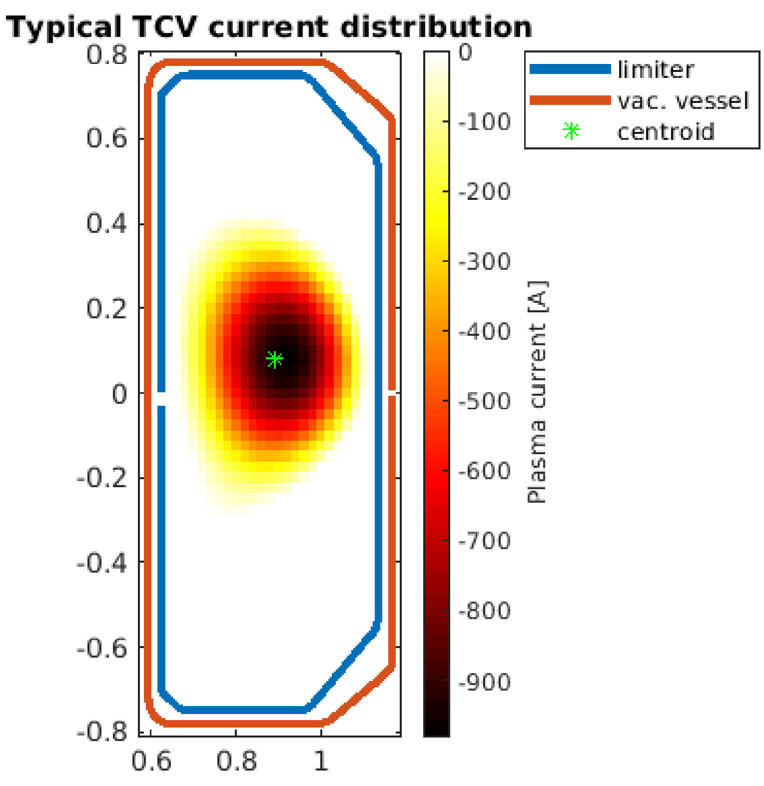
```
clear;
fromfile = true; % set true to load from file
fname = 'ex2_data.mat';
if fromfile && ~isempty(which(fname))
  load('ex2_data.mat'); % if file exists, load it
  disp('loaded TCV data from file');
else
  % Loading an equilibrium from TCV
  [L,~,LY] = fbt('tcv',61400,1,'selu','v','izgrid',1);
  % uncomment this to save file
  % save(fname,'L','LY');
  disp('file not found, loaded from fbt');
end
```

```
loaded TCV data from file
```

Define plasma distribution on the x grid

```
Ix  = zeros(L.nzx,L.nrx);
Ix(2:end-1,2:end-1) = LY.Iy;

% determine current distribution centroid position
```

```
r0  = sum(L.rrx(:).*Ix(:))/sum(Ix(:));
z0  = sum(L.zzx(:).*Ix(:))/sum(Ix(:));
Ip0 = sum(Ix(:)); % calculating the total current

% plot plasma current distribution inside TCV vac. vessel
figure(1); clf;
imagesc(L.rx,L.zx,Ix); %plots current distribution
axis tight; axis xy equal;
hold on;
plot(L.G.rl,L.G.zl,'LineWidth',3);  % plot limiter: plasma facing
 tiles
plot(L.G.rv,L.G.zv,'LineWidth',3);  % plot vacuum vessel behind the
 limiter
plot(r0,z0,'g*');     % plot current centroid determined above
cb = colorbar; colormap('hot');
cb.Label.String = 'Plasma current [A]';
legend('limiter','vac.
 vessel','centroid', 'Location', 'northeastoutside');
title('Typical TCV current distribution');
```



Compute expected measurements for the Ix plasma distribution

```
Ia  = LY.Ia;   % pull active coil current values. No noise
% measurements = plasma current + active coil current contributions
```

```
Bm  = L.G.Bmx*Ix(:) + L.G.Bma*Ia; % magnetic probes measurements
Ff  = L.G.Mfx*Ix(:) + L.G.Mfa*Ia; % flux loops measurements
```

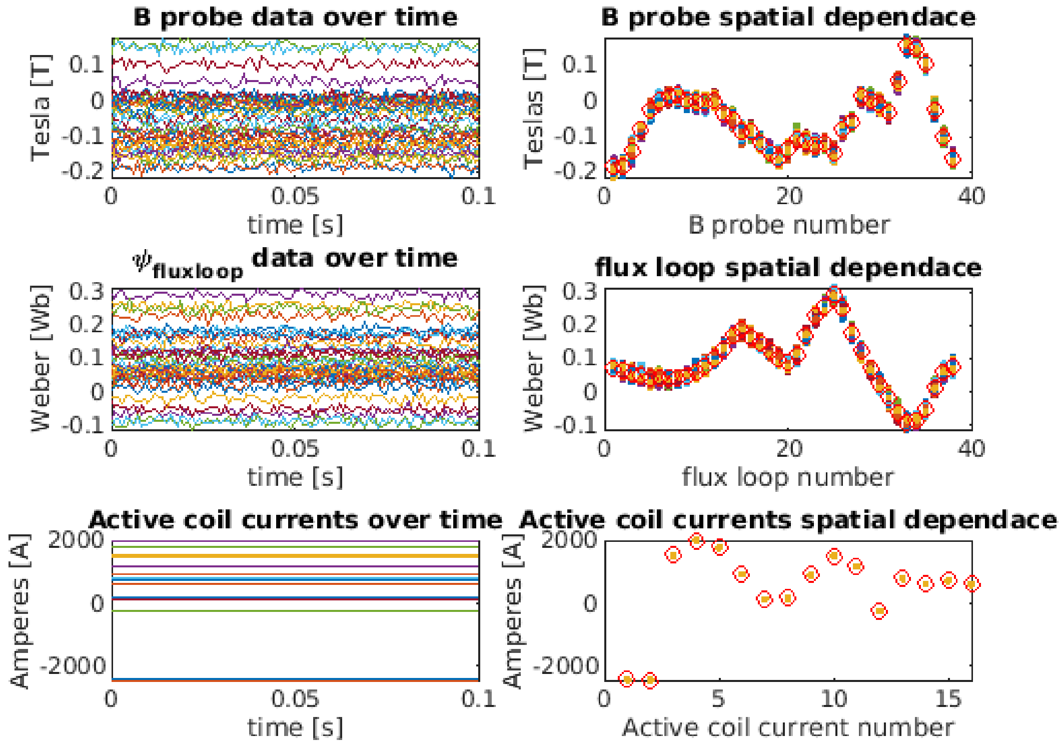
Add noise to measurements and turn into time-dependant signal

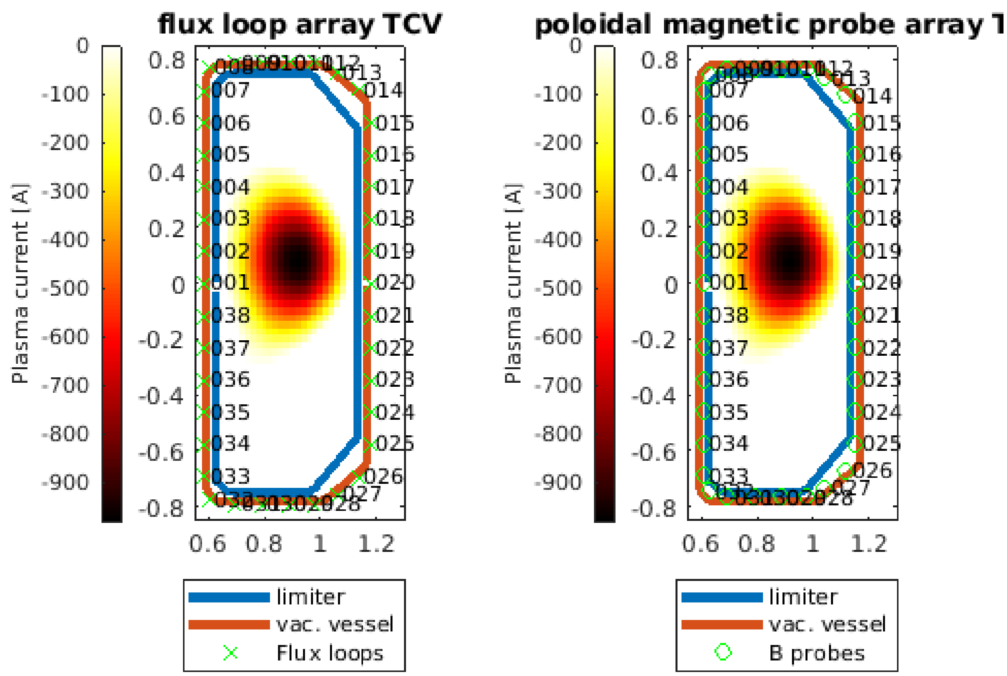
```
eBm = 10e-3;  % 10mT noise
eFf = 10e-3;  % 10mWb noise
eIa = 0;      % no error on Ia

% time grid
time = 0:1e-3:0.1;
nt = numel(time);
% time evolution
Bm_meas = Bm + eBm*randn(numel(Bm),nt);
Ff_meas = Ff + eFf*randn(numel(Ff),nt);
Ia_meas = Ia + eIa*randn(numel(Ia),nt);
```

Plot measurment data and measurment locations

```
figure(2); clf;
% magnetic probes time evolution
subplot(321)
plot(time,Bm_meas);
title('B probe data over time'); xlabel('time [s]'); ylabel('Tesla
 [T]');
% flux loops time evolution
subplot(323)
plot(time,Ff_meas);
title('\psi_{fluxloop} data over time'); xlabel('time [s]');
 ylabel('Weber [Wb]');
subplot(325)
plot(time,Ia_meas);
title('Active coil currents over time'); xlabel('time [s]');
 ylabel('Amperes [A]');


% measurements per magnetic probe
subplot(322)
plot(1:numel(Bm),Bm_meas,'.',1:numel(Bm),Bm,'or')
title('B probe spatial dependace')
xlabel('B probe number'); ylabel('Teslas [T]');

% measurements per flux loop
subplot(324)
plot(1:numel(Ff),Ff_meas,'.',1:numel(Ff),Ff,'or');
title('flux loop spatial dependace')
xlabel('flux loop number'); ylabel('Weber [Wb]');

% measurements of active coil currents
```

```
subplot(326)
plot(1:numel(Ia), Ia_meas,'.',1:numel(Ia), Ia,'or');
title('Active coil currents spatial dependace')
xlabel('Active coil current number'); ylabel('Amperes [A]');


% Plot the positons of the magentic probes and flux loops
figure(3); clf;
subplot(121)
imagesc(L.rx,L.zx,Ix);  axis tight; axis xy equal;
cb = colorbar('westoutside'); colormap('hot');
cb.Label.String = 'Plasma current [A]';
hold on;
plot(L.G.rl,L.G.zl,'LineWidth',3);
plot(L.G.rv,L.G.zv,'LineWidth',3);
plot(L.G.rf,L.G.zf,'gx');
text(L.G.rf+0.02,L.G.zf,L.G.dimf, 'FontSize', 8);
legend('limiter','vac. vessel','Flux loops', ...
        'Location', 'southoutside');
title('flux loop array TCV');
xlim([0.55, 1.3]); ylim([-0.85, +0.85]);

subplot(122)
imagesc(L.rx,L.zx,Ix); axis tight; axis xy equal;
cb = colorbar('westoutside'); colormap('hot');
cb.Label.String = 'Plasma current [A]'; hold on;
plot(L.G.rl,L.G.zl,'LineWidth',3);
plot(L.G.rv,L.G.zv,'LineWidth',3);
plot(L.G.rm,L.G.zm,'go')
text(L.G.rm+0.02,L.G.zm,L.G.dimm, 'FontSize', 8);
legend('limiter','vac. vessel','B probes', ...
      'Location', 'southoutside');
title('poloidal magnetic probe array TCV');
xlim([0.55, 1.3]); ylim([-0.85, +0.85]);
```


B probe data over time
Tesla [T]
time [s]
B probe spatial dependace
Teslas [T]
B probe number
$\psi_{fluxloop}$ data over time
Weber [Wb]
time [s]
flux loop spatial dependace
Weber [Wb]
flux loop number
Active coil currents over time
Amperes [A]
time [s]
Active coil currents spatial dependace
Amperes [A]
Active coil current number


flux loop array TCV
Plasma current [A]
limiter
vac. vessel
Flux loops
poloidal magnetic probe array T
Plasma current [A]
limiter
vac. vessel
B probes

Define new grid boundaries of representation with fewer elements (2x3)

Split the x grid into fewer rectangular regions and let the current distribution be constant in that region.

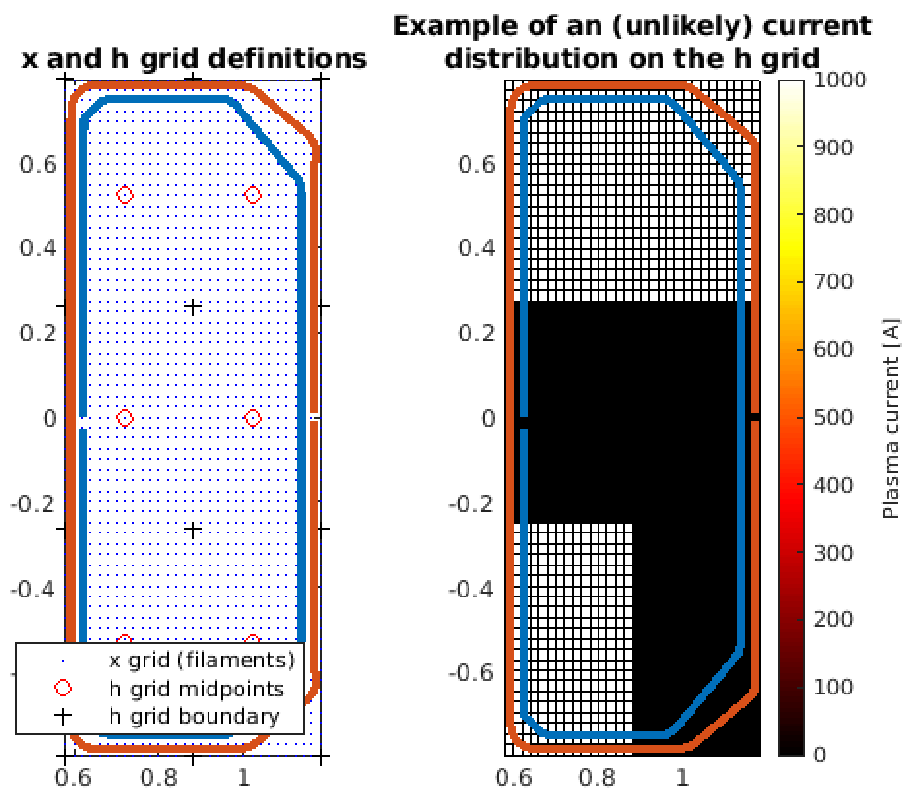
```
nr = 2; % number of radial regions
nz = 3; % number of vertical regions
nh = nr*nz; % total number of degrees of freedom
rhgrid = linspace(min(L.G.rx),max(L.G.rx),nr+1);
zhgrid = linspace(min(L.G.zx),max(L.G.zx),nz+1);
```

Build transformation matrix T_{xh}

let 'x' be the full (original) grid and 'h' be the reduced representation Let I_x = T_{xh}*I_h. Build this matrix T_{xh} here:

```
Txh = zeros(L.nx, nh);
[rrx,zzx] = meshgrid(L.G.rx,L.G.zx);
for ir = 1:nr
    for iz = 1:nz
        ii = (ir-1)*(nz) + iz; % element index
        % choose region bounded by points defining borders of the
 rectangles
        txy = (rhgrid(ir) <= rrx) & (rrx < rhgrid(ir+1))...
            & (zhgrid(iz) <= zzx) & (zzx < zhgrid(iz+1)) ;
        Txh(:,ii) = txy(:);
    end
end
% calculate mean r,z of each new grid point for illustration purposes
 later
rhmean = sum(Txh.*rrx(:),1)./sum(Txh,1);
zhmean = sum(Txh.*zzx(:),1)./sum(Txh,1);

% plot grids
figure(4); clf;
subplot(121);
p   = plot(rrx,zzx,'.b','markersize',2); hold on;
p1  = plot(rhmean,zhmean,'or');
axis equal tight;
[rhh,zhh] = meshgrid(rhgrid,zhgrid); hold on;
p2 = plot(rhh,zhh,'+k');
title('x and h grid definitions');
hold on; % plot limiter and vac vessel
plot(L.G.rl,L.G.zl,'LineWidth',3);
plot(L.G.rv,L.G.zv,'LineWidth',3);
legend([p(1) p1(1) p2(1)],...
'x grid (filaments)','h grid midpoints','h grid boundary',...
'location','south');

% example of a (unlikely) current distribution in the smaller h-grid
 mapped to x-grid
```

```
Ih    = zeros(nh,0);
Ih([1,3,6],1) = 1e3;  % imaginary current distribution 1e3 in boxes 1,
 3, and 6
Ix_ex = Txh*Ih;
subplot(122)
pcolor(L.G.rx,L.G.zx,reshape(Ix_ex,L.nzx,L.nrx));  axis equal tight;
cb = colorbar; colormap('hot');
cb.Label.String = 'Plasma current [A]'; hold on;
title({'Example of an (unlikely) current', 'distribution on the h
 grid'});
hold on; % plot limiter and vac vessel
plot(L.G.rl,L.G.zl,'LineWidth',3);
plot(L.G.rv,L.G.zv,'LineWidth',3);
```



a) Least-squares problem for current distribution on the full grid

Write a least-squares problem to determine the current distribution I_x on the full x grid from the measurements of magnetic probes and flux loops, in the form $\min ||AI_x - b||_2$. Does this WLS problem have a unique solution?

```
% $\min||AI_x - b||_2$.- find A and b.
% b   = [ Bm - L.G.Bma*Ia ; Ff - L.G.Mfa*Ia];
% A   = [ L.G.Bmx ; L.G.Mfx];
```

```
% can't do because the number of measurements is not enough
% you have 38 Bms and 38 Ffs, but you are trying to solve for values
 of
% currents in a space of 65x31 = 2015. This does not have a unique
 solution.
% There are many ways to fit 2015 Ix spots from 76 experimental
 points.
```

b) Least-squares problem for current distribution at reduced h-grid

Write a least-squares problem to determine the current distribution I_h on the reduced h grid from the measurements. What are A and b? Does this WLS problem have a unique solution? Write also the weighted least-squares problem definition.

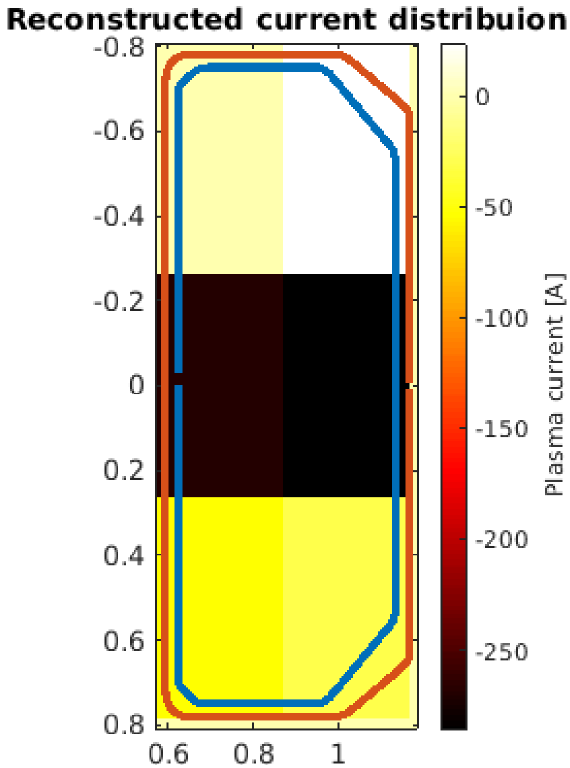
```
% normal least squares: Ah*Ih - b
b   = [ Bm - L.G.Bma*Ia ; Ff - L.G.Mfa*Ia];
A   = [ L.G.Bmx ; L.G.Mfx];
Ah  = [ L.G.Bmx*Txh; L.G.Mfx*Txh];

% now, you would be getting solutions for size(Ih) = 6 from 38+38 = 76
% data points. The solution now can be found analytically and it is
 unique.
```

c) Solution of the least-squares problem

Write the solution of least-squares problem in (b) in the form $I_h = H \cdot b$. How can H be determined from A? Plot Ih from the first time slice of measurements

```
b   = [ Bm_meas(:,1) - L.G.Bma*Ia ; Ff_meas(:,1) - L.G.Mfa*Ia];
% vector of measurements from first time slice of noisy measurements

% least squares
H   = inv(Ah'*Ah)*Ah'; % H = A+ (see lecture slides)
Ih  = H*b;
% could have also used Ih = lsqrt(Ah,b)
% could have also used H = pinv(Ah);

% weighted least squares
% first define diagnoal weight matrix
W   = diag([ones(length(Bm),1)./(eBm^2) ; ones(length(Ff),1)./
(eFf^2)]);
% WLS then:
HW  = inv(Ah'*W*Ah)*Ah'; % or HW = pinv(W*Ah);
IhW = HW*(W*b);

% plot the current distribution obtained. Use Ix version of Ih for
 plotting
Ixhat = Txh*IhW;
figure(5); clf;
imagesc(L.rx, L.zx, reshape(Ixhat, L.nzx, L.nrx));
```

```
cb = colorbar; colormap('hot');
cb.Label.String = 'Plasma current [A]'; hold on;

axis equal; axis tight; hold on;
title('Reconstructed current distribuion');
hold on; % plot limiter and vac vessel
plot(L.G.rl,L.G.zl,'LineWidth',3);
plot(L.G.rv,L.G.zv,'LineWidth',3);
```



d) Write the weighted least squares solution in (c) in the form Ih = Ahy*y

where y = [Bm; Ff; Ia] see slide 28 for graphical matrix manipulation

```
y     = [Bm_meas(:,1) ; Ff_meas(:,1); Ia_meas(:,1)];
Q     = HW*W;
Ahy   = [Q , -Q*[L.G.Bma; L.G.Mfa]];
IhW_y = Ahy*y;

% check that the new IhW_y is the same as IhW above
% figure; imagesc(IhW_y-IhW); colorbar;
```

e) Write a linear estimator for the plasma current

Recall that the plasma current can be determined from the current distribution as $I_p = \sum_{j=1}^{n_x} I_{x,j}$. Write a linear estimator for the plasma current from the measurements such that $I_p = H_{I_p} \cdot y$.

```
H_Ip = ones(1,size(Txh,1))*Txh*Ahy;
Ip   = H_Ip*y;

fprintf('Estimate Ip: %0.3f kA vs actual Ip input: %0.3f kA \n', ...
        Ip/1e3, sum(Ix(:))/1e3);
```

```
Estimate Ip: -193.524 kA vs actual Ip input: -190.000 kA
```

f) Estimate of r and z based on the measurements

We can define an estimator for the plasma radial and vertical position, by assuming the position corresponds to the centroid of the current distribution:

$\hat{r} = \frac{\sum_j^{n_x} r_{x,j} I_{x,j}}{\sum_i^{n_x} I_{x,i}}$ and $\hat{z} = \frac{\sum_j^{n_x} z_{x,j} I_{x,j}}{\sum_i^{n_x} I_{x,i}}$

Use these experessions to compute an estimate of r and z from the measurement vector y. Is this a linear estimator?

```
rIn   = repmat(L.rx, size(L.zx))'; % matrix of radial positions in x
 space
H_rIp = (rIn*(Txh*Ahy));
rIp   = H_rIp*y;
rEst  = (H_rIp*y)/(H_Ip*y);
fprintf('Estimate r: %0.3f [m] vs actual r0: %0.3f [m] \n', ...
        rEst, r0);

zIn   = repmat(L.zx, size(L.rx))'; % matrix of vertical positions in x
 space
H_zIp = (zIn*(Txh*Ahy));
zIp   = H_zIp*y;
zEst  = (H_zIp*y)/(H_Ip*y);
fprintf('Estimate z: %0.3f [m] vs actual z0: %0.3f [m] \n', ...
        zEst, z0);
% operators are clearly nonlinear on y
```

```
Estimate r: 0.880 [m] vs actual r0: 0.890 [m]
Estimate z: 0.092 [m] vs actual z0: 0.081 [m]
```

g) Estimator for the plasma rI_p and zI_p

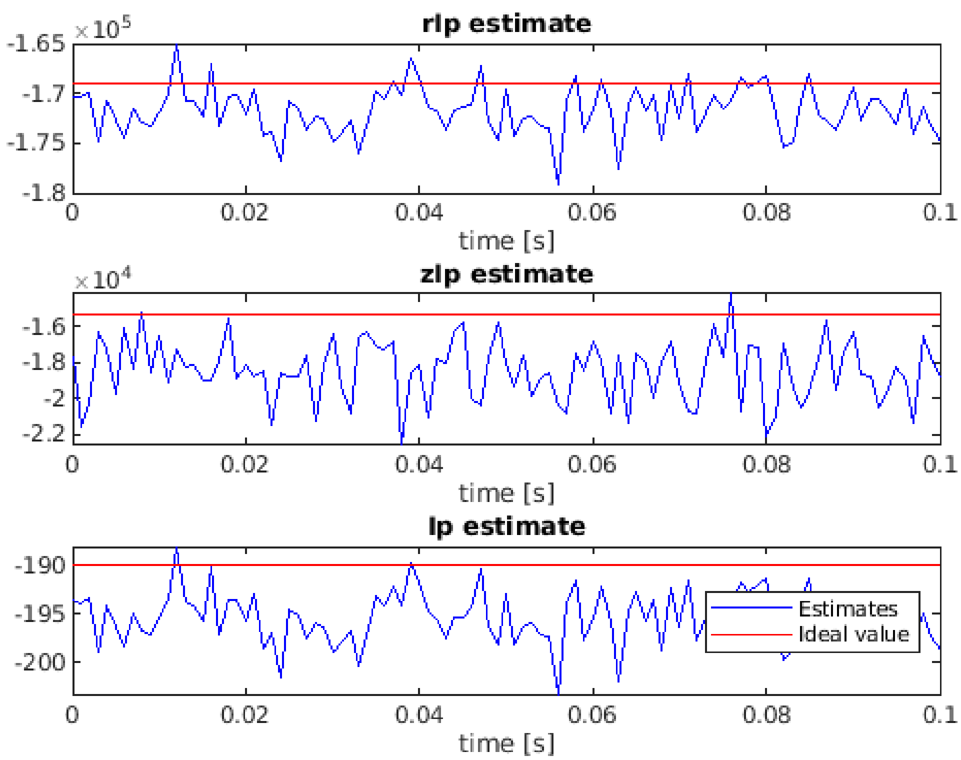
Write a estimator for the plasma rI_p and zI_p i.e. the product of radial/vertical position and plasma current. Is this linear?

```
rIp   = H_rIp*y;
zIp   = H_zIp*y;

% these are linear :-)
```

h) Outputs of the linear estimators

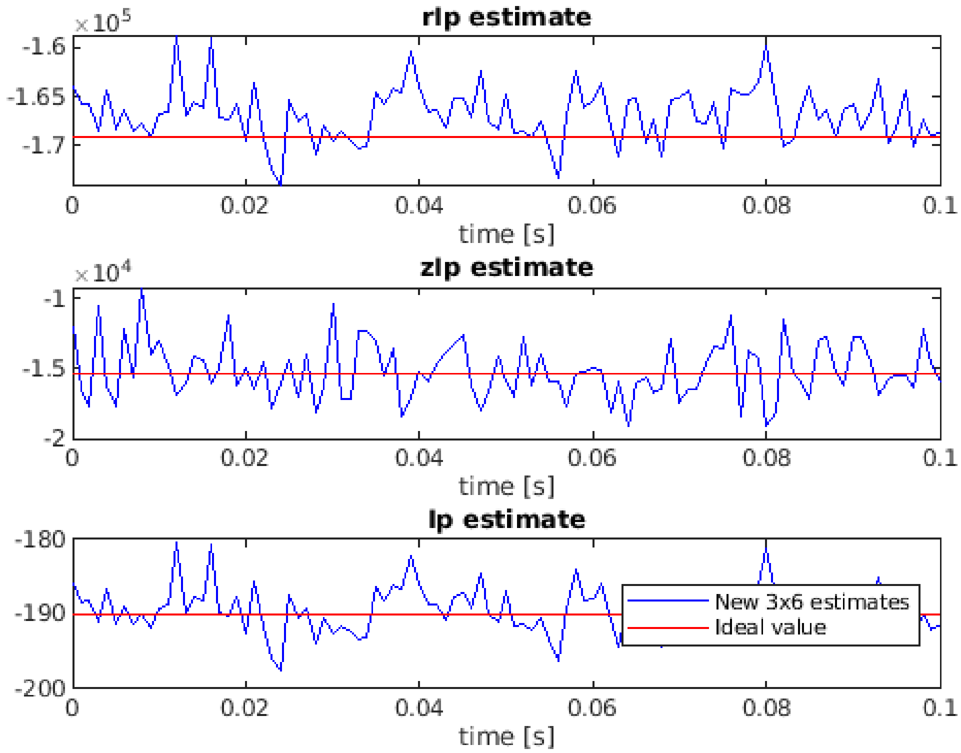
Plot the outputs of the linear estimators on each of the measurement data points and compare to the known values of $r_0 I_{p0}, z_0 I_{p0}, I_{p0}$. Improve the estimate of $z_0 I_{p0}$ and % I_{p0} and explain your solution.

```
y_time  = [Bm_meas; Ff_meas; Ia_meas];
Ip_est  = H_Ip*y_time;
rIp_est = H_rIp*y_time;
zIp_est = H_zIp*y_time;

figure(6); clf;
subplot(311);
plot(time,rIp_est,'b',time,r0*Ip0*ones(numel(time),1),'r')
title('rIp estimate'); xlabel('time [s]');
subplot(312);
plot(time,zIp_est,'b',time,z0*Ip0*ones(numel(time),1),'r')
title('zIp estimate'); xlabel('time [s]');
subplot(313);
plot(time,Ip_est./1e3,'b',time,Ip0*ones(numel(time),1)./1e3,'r')
title('Ip estimate'); xlabel('time [s]');
legend('Estimates', 'Ideal value', 'Location', 'east');
```



explore effects of nr/nz

```
% try a different discretization
nr = 3; % number of radial regions
nz = 6; % number of vertical regions
nh = nr*nz; % total number of degrees of freedom
rhgrid = linspace(min(L.G.rx),max(L.G.rx),nr+1);
zhgrid = linspace(min(L.G.zx),max(L.G.zx),nz+1);
% re-Build transformation matrix T_{xh}
Txh2 = zeros(L.nx, nh);
[rrx,zzx] = meshgrid(L.G.rx,L.G.zx);
for ir = 1:nr
    for iz = 1:nz
        ii = (ir-1)*(nz) + iz; % element index
        % choose region bounded by points defining borders of the
 rectangles
        txy = (rhgrid(ir) <= rrx) & (rrx < rhgrid(ir+1))...
            & (zhgrid(iz) <= zzx) & (zzx < zhgrid(iz+1)) ;
        Txh2(:,ii) = txy(:);
    end
end


% recalculate estimators due to change in Txh
Ah2    = [ L.G.Bmx*Txh2; L.G.Mfx*Txh2];
HW2    = pinv(W*Ah2);
Q2     = HW2*W;
Ahy2   = [Q2 , -Q2*[L.G.Bma; L.G.Mfa]];
H_Ip2  = ones(1,size(Txh2,1))*Txh2*Ahy2;
H_rIp2 = (rIn*(Txh2*Ahy2));
H_zIp2 = (zIn*(Txh2*Ahy2));

% replot results to see effect
Ip_est  = H_Ip2*y_time;
rIp_est = H_rIp2*y_time;
zIp_est = H_zIp2*y_time;

figure(7); clf;
subplot(311);
plot(time,rIp_est,'b',time,r0*Ip0*ones(numel(time),1),'r')
title('rIp estimate'); xlabel('time [s]');
subplot(312);
plot(time,zIp_est,'b',time,z0*Ip0*ones(numel(time),1),'r')
title('zIp estimate'); xlabel('time [s]');
subplot(313);
plot(time,Ip_est./1e3,'b',time,Ip0*ones(numel(time),1)./1e3,'r')
title('Ip estimate'); xlabel('time [s]');
legend('New 3x6 estimates', 'Ideal value', 'Location', 'east');

% Clearly the new estimates are better: more discretization points
% lead to a better Ip, rIp, and zIp estimates.
```



i) (optional) Effect of nh

Plot the mean and standard deviation of I_p, zI_p, rI_p estimates for various values of nr, nz (and $nh = nz * nr$). Does the estimator quality increase, decrease or stay the same when increasing nh? Explain why.

j) (optional) Effect on Circuit current measurements

Now assume that the circuit current measurements \texttt{Ia_meas} is also affected by measurement noise of standard deviation \texttt{eIa=100[A]}. Check the quality of your estimator and compare it to the non-noisy case. Reformulate the least squares problem to jointly estimate the plasma current distribution and circuit currents from the measurements and compare the estimation results.

Published with MATLAB® R2020a