Exercise - Control and operation of tokamaks 2023

Topic: Free boundary evolution, FBT & LIUQUE reconstruction

If access to TCV data using MDSplus is impossible use MAT-files by setting use_files = true

```
use_files = true;
```

For generating .pdf figures outputs select print_figures = true

```
print_figures = false;
```

Exercise 1 - Design of the vacuum equilibrium field

```
% Input equilibrium parameters
Ip = 400e3;     % [A] total plasma current
kappa = 1.6;    % Elongation
betap = 0.3;    % Poloidal beta
li = 1;         % internal inductance
a = 0.25;       % [m] Plasma minor radius
r0 = 0.87;      % [m] R position of plasma axis
z0 = 0;         % [m] Z position of the plasma axis
```

Required parameters

```
n = -1.44;      % field index
Br0 = 0;        % Br component on axis
```

Compute required contraints Bz component on axis

```
Bz0 = -mu0*Ip/4/pi/r0*(log(8*r0/a/sqrt(kappa)) + betap + li/2 - 3/2);
% Derivative of Bz componet on axis
dBz0dr = -n * Bz0 / r0;
```

Green's function's from function exgf

```
if use_files
  prefix = 'tcv_102735_fbt';
  fileargs = {'pfile',[prefix,'_P'],'gfile',[prefix,'_G'],...
      'xfile',[prefix,'_X']};
  L = fbt('file',102735,[],fileargs{:});
else
  L = fbt('tcv',102735);
end
[Mxa,Brxa,Bzxa,~,Brpa,Bzpa,Cpa,na] = exgf(L.G,r0,z0);
rx = L.G.rx; zx = L.G.zx;
nr = numel(rx); nz = numel(zx);
```

1.a Solve LSQ problem

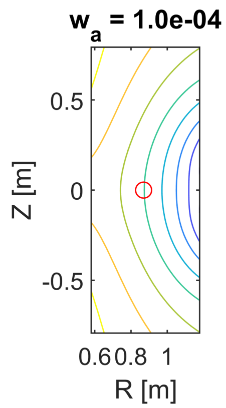
The minimization problem can be brought in the form $\min_x ||Ax - b||_2^2$, as can be seen below in the LSQ setup.

The weight wa measures the relative importance in the cost function of respecting the magnetic field/curvature against minimizing the current amplitude: higher weight means smaller current but worse matching of the B/curvature requirements. Set here the current weights for LS minimization and vary to analyse their impact.

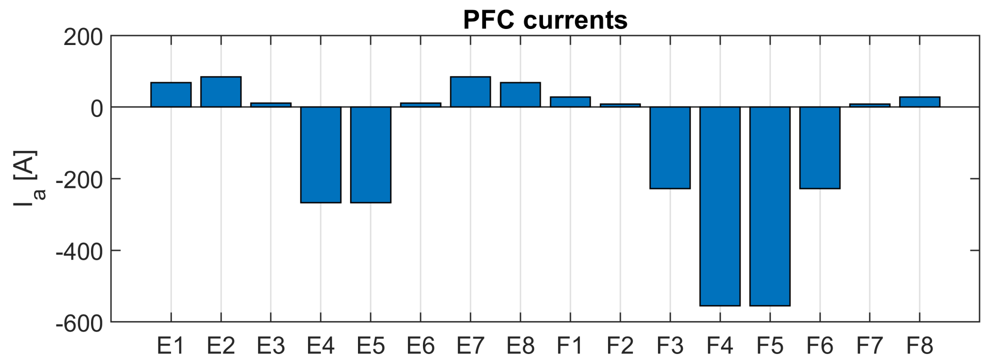
```
wa = [1e-4,1e-6];
```

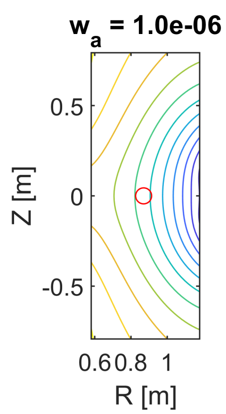
Set up LSQ problem:

```
for ii = 1:length(wa)
    A = [Brpa ; Bzpa; Cpa; eye(na)*wa(ii)];
    b = [Br0; Bz0; dBz0dr; zeros(na,1)];

    % Solve using Matlab \ operator, this is equivalent of solving (A'A) Ia = A'b
    Ia_lsq = A\b;

    % Obtain magnetic field results
    Brq_lsq = Brpa*Ia_lsq;
    Bzq_lsq = Bzpa*Ia_lsq;
    nBq_lsq = -r0 / Bz0*(Cpa*Ia_lsq);

    % Obtain flux, reshape on rz-grid
    psi_lsq = reshape(Mxa* Ia_lsq, [nz,nr]);

    % Plot results:
    fig = figure;
    subplot(2,2,1) % flux map
    contour(rx,zx,psi_lsq); hold on; plot(r0,z0,'ro');
    title(sprintf('w_a = %1.1e',wa(ii)) ), xlabel('R [m]'), ylabel('Z [m]');
    axis equal

    text(1.75, 0, sprintf(' Br requested= %5.2f, obtained=%5.2f \n Bz requested=%5.2f, obtained
      Br0,Brq_lsq,Bz0,Bzq_lsq,n,nBq_lsq))

    subplot(2,2,[3,4]) % coil currents
    bar(Ia_lsq); title('PFC currents');
    ylabel('I_a [A]');
    set(gca,'xtick',1:16,...
        'xticklabel',[num2str((1:8)','E%d');num2str((1:8)','F%d')],'xgrid','on');

    if print_figures == true
        print_pdf(fig, sprintf('lsq_wa%1.1e.pdf',wa(ii)));
    end
end
```

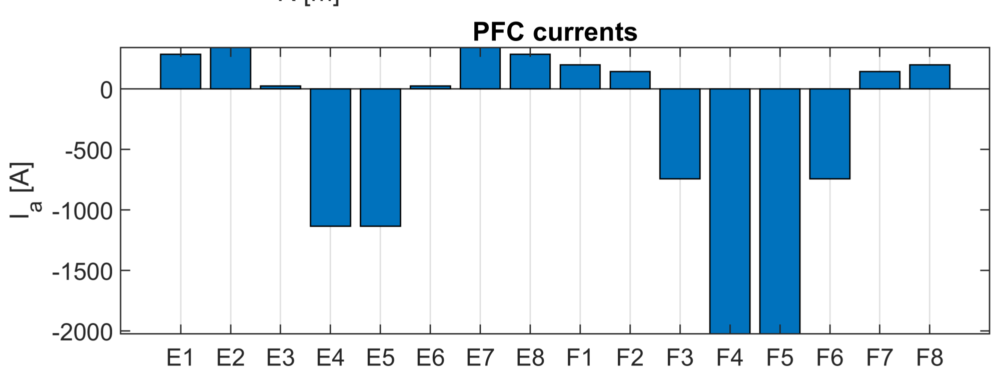


Br requested= 0.00, obtained=-0.00
Bz requested=-0.11, obtained=-0.03
nB requested=-1.44, obtained=-0.37





Br requested= 0.00, obtained= 0.00
Bz requested=-0.11, obtained=-0.11
nB requested=-1.44, obtained=-1.44



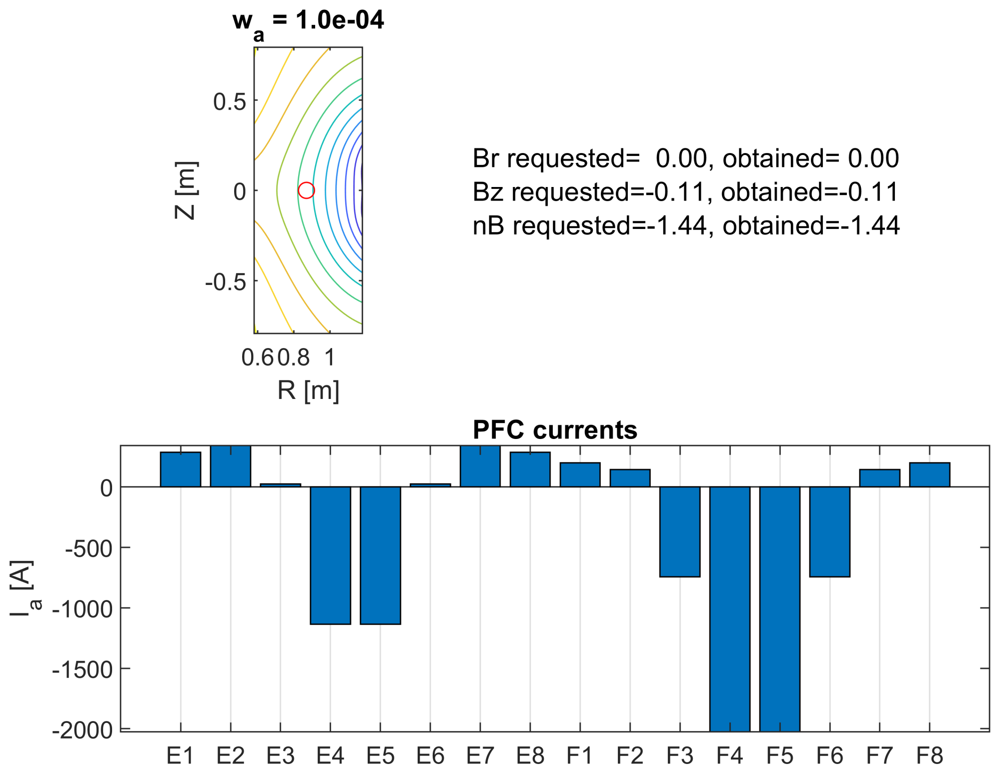
1.b Solve LSQ with fixed contraints

The solution is not influenced by wa. Changing wa will change the minium value of the cost function but not the value of the minimizer. Since the fields are now imposed as constraints, the currents obtained will be high, as in the previous exercise when small wa is used. Set here the current weights and vary them to analyse their impact:

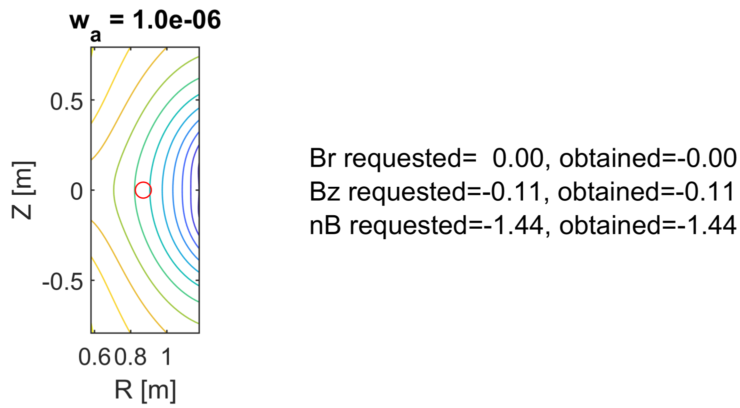
```
wa = [1e-4,1e-6];
```

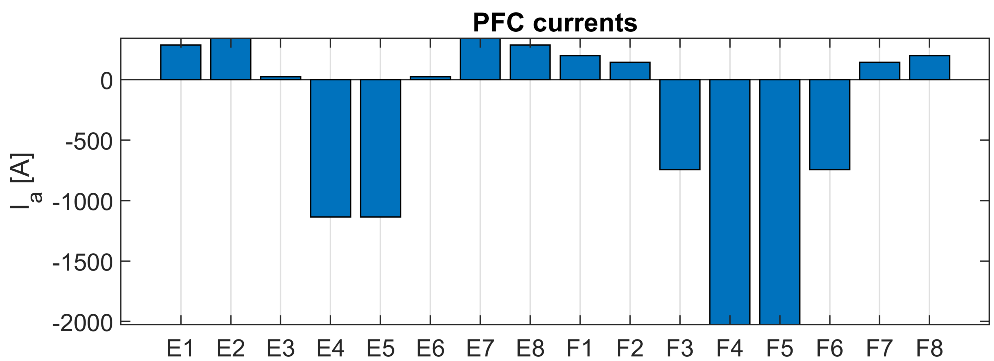
```
for ii = 1:length(wa)
    % Set up problem with fixed constraints
    Aeq = [Brpa;Bzpa;Cpa];
    beq = [Br0; Bz0; dBz0dr];
    A= wa(ii)*eye(na);
    b = zeros(na,1);

    % Solve using lsqlin Matlab function. This is equivalent of minimizing the
    % functional imposing the constraints with Lagrangian multiplier.
    Ia_fixed = lsqlin(A,b,[],[],Aeq,beq);

    % Obtain magnetic results
    Brq_fixed = Brpa*Ia_fixed;
    Bzq_fixed = Bzpa*Ia_fixed;
    nBq_fixed = -r0 / Bz0*(Cpa*Ia_fixed);

    % Obtain flux, reshape on rz-grid
    psi_fixed = reshape(Mxa* Ia_fixed, [nz,nr]);

    % Plot results:
    fig = figure;
    subplot(2,2,1) % flux map
    contour(rx,zx,psi_fixed); hold on; plot(r0,z0,'ro');
    title(sprintf('w_a = %1.1e',wa(ii)) ), xlabel('R [m]'), ylabel('Z [m]'), axis equal

    text(1.75, 0, sprintf(' Br requested= %5.2f, obtained=%5.2f \n Bz requested=%5.2f, obtaine
      Br0,Brq_fixed,Bz0,Bzq_fixed,n,nBq_fixed))

    subplot(2,2,[3,4]) % coil currents
    bar(Ia_fixed);
    title('PFC currents'); ylabel('I_a [A]');
    set(gca,'xtick',1:16,...
        'xticklabel',[num2str((1:8)','E%d');num2str((1:8)','F%d')],'xgrid','on');

    if print_figures == true
        print_pdf(fig, sprintf('fixed_constraints_wa%1.1e.pdf',wa(ii))); end
end
```

```
Minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.

<stopping criteria details>
```




```
Minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.

<stopping criteria details>
```





c. Solve using FBT

```
if use_files
  prefix = 'tcv_102735_fbt';
  fileargs = {'pfile',[prefix,'_P'],'gfile',[prefix,'_G'],'xfile',[prefix,'_X']};
  [L,~,LY] = fbt('file',102735,[],'debug',1,fileargs{:});
else
  [L,~,LY] = fbt('tcv' ,102735,[],'debug',1); % Run FBT
end
```

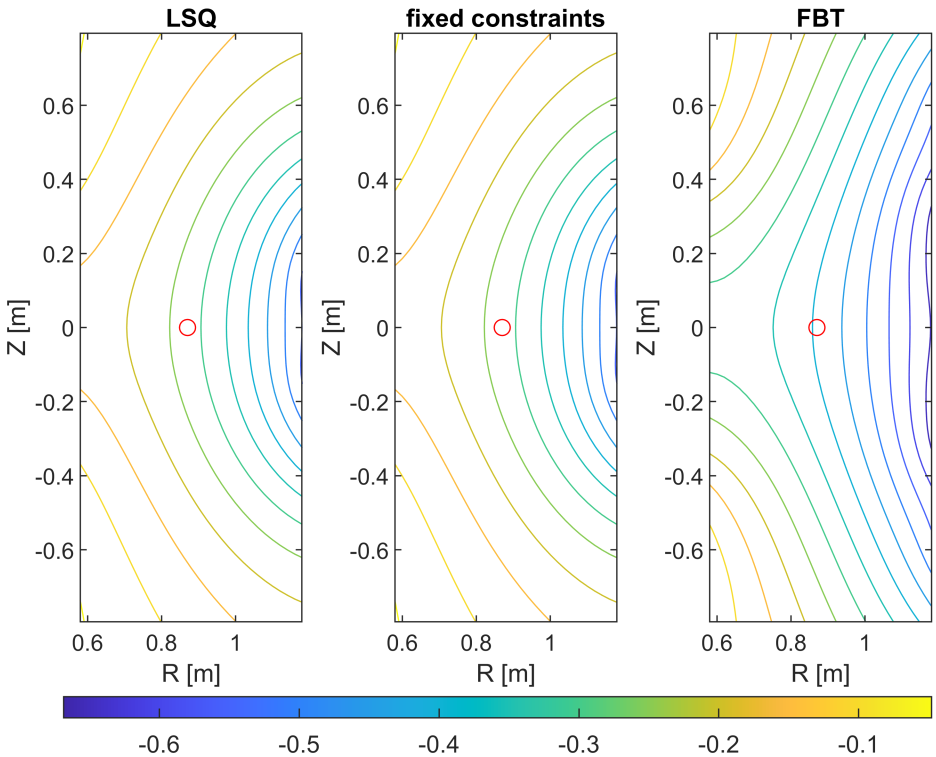
```
TCV#102735 0.0100s/01/01 Ip=-400kA,nA=1,res=      NaN,tol=2.00e-04,r,z=0.885,-0.000m
TCV#102735 0.0100s/02/01 Ip=-400kA,nA=1,res=2.93e-01,tol=2.00e-04,r,z=0.882,-0.000m
TCV#102735 0.0100s/03/01 Ip=-400kA,nA=1,res=3.40e-02,tol=2.00e-04,r,z=0.881,-0.000m
TCV#102735 0.0100s/04/01 Ip=-400kA,nA=1,res=9.16e-03,tol=2.00e-04,r,z=0.881,-0.000m
TCV#102735 0.0100s/05/01 Ip=-400kA,nA=1,res=2.68e-03,tol=2.00e-04,r,z=0.880,-0.000m
TCV#102735 0.0100s/06/01 Ip=-400kA,nA=1,res=7.92e-04,tol=2.00e-04,r,z=0.880,-0.000m
TCV#102735 0.0100s/07/01 Ip=-400kA,nA=1,res=2.35e-04,tol=2.00e-04,r,z=0.880,-0.000m
TCV#102735 0.0100s/08/01 Ip=-400kA,nA=1,res=6.99e-05,tol=2.00e-04,r,z=0.880,-0.000m
```

```
% Obtain magnetic field results
Brq_fbt = Brpa*-LY.Ia;
Bzq_fbt = Bzpa*-LY.Ia;
nBq_fbt = -r0 / Bz0*(Cpa*-LY.Ia);

% Obtain flux, reshape on rz-grid
psi_fbt = reshape(Mxa*-LY.Ia, [nz,nr]); % reshape on grid
```

Comparison of flux maps obtained by the different methods.

Note, that the field curvatures are similar, but the FBTE field is also designed to yield the desired shape matching in Grad-Shavranof equation, while no such constraint was imposed in the vacuum calculation.

```
% Define minimum and maximum values for colorbar
cax_max = max([max(psi_fbt(:)),max(psi_fixed(:)),max(psi_lsq(:))]);
cax_min = min([min(psi_fbt(:)),min(psi_fixed(:)),min(psi_lsq(:))]);
psi_grid = -1:0.05:1;
fig = figure;
s_lsq = subplot(1,3,1); % LSQ
contour(s_lsq,rx,zx,psi_lsq, psi_grid); hold on; plot(r0,z0,'ro');
title(s_lsq,'LSQ'), xlabel(s_lsq,'R [m]'), ylabel(s_lsq,'Z [m]'); axis equal
caxis(s_lsq,[cax_min,cax_max]);

s_fixed = subplot(1,3,2); % LSQ with fixed constraints
contour(s_fixed,rx,zx,psi_fixed, psi_grid); hold on; plot(r0,z0,'ro');
title(s_fixed,'fixed constraints'), xlabel(s_fixed,'R [m]'), ylabel(s_fixed,'Z [m]');
axis equal
caxis(s_fixed,[cax_min,cax_max]);

s_fbt = subplot(1,3,3); % FBTE
contour(s_fbt,rx,zx,psi_fbt,psi_grid); hold on; plot(r0,z0,'ro')
title(s_fbt,'FBT'), xlabel(s_fbt,'R [m]'), ylabel(s_fbt,'Z [m]');
axis equal
caxis(s_fbt,[cax_min,cax_max]);

% Setup colorbar
h = axes(fig,'visible','off'); c = colorbar();
caxis(h,[cax_min,cax_max]);
c.Location = 'southoutside';
c.Position(2) = 0.35*c.Position(2); c.Position(4) = 0.5*c.Position(4);
```




```
if print_figures == true
    print_pdf(fig,'comparison_1c'); end
```

1.d Compare flux contributions

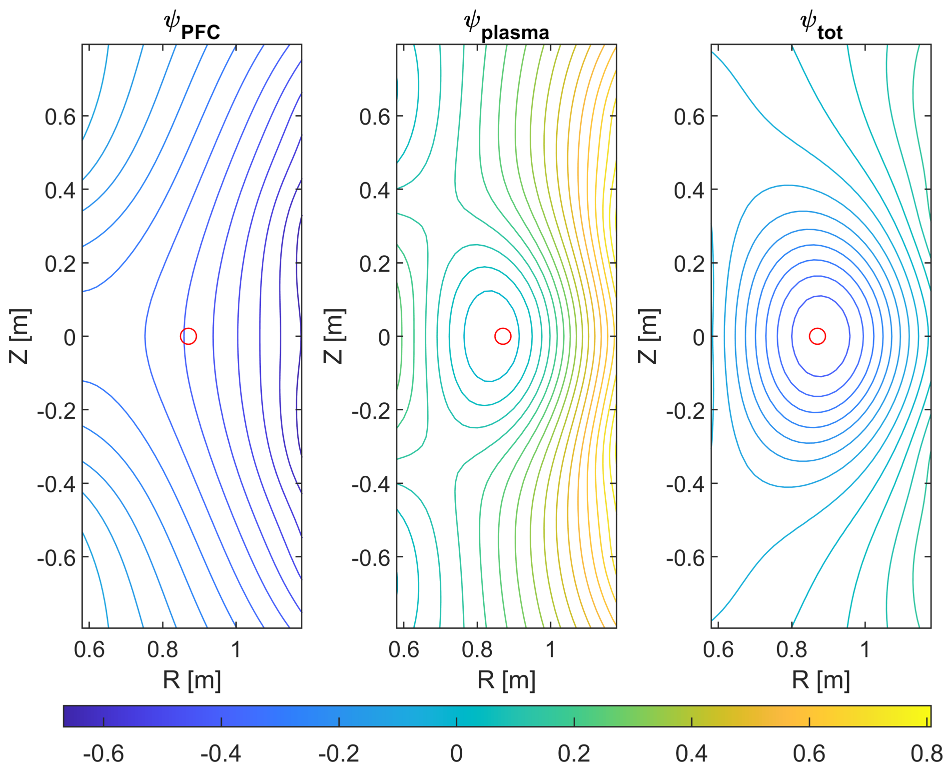
Comparison of flux obtained using fbt of (i) currents in the PF coils, (ii) plasma current, (iii) total flux. The vacuum poloidal flux actually acts to compress the flux at the low field side of the tokamak, in order to to balance the outward radial force of the plasma.

```
psi_tot = LY.Fx;
psi_coils = reshape(Mxa*-LY.Ia, [nz,nr]);
psi_plasma = psi_tot-psi_coils;
psi_grid = -1:0.05:1,
```

```
psi_grid = 1×41
   -1.0000   -0.9500   -0.9000   -0.8500   -0.8000   -0.7500   -0.7000   -0.6500 ···
```

```
% Define minimum and maximum values for colorbar
cax_max = max([max(psi_tot(:)),max(psi_coils(:)),max(psi_plasma(:))]);
cax_min = min([min(psi_tot(:)),min(psi_coils(:)),min(psi_plasma(:))]);

fig = figure;
s_fbt = subplot(1,3,1); % PFC flux
contour(s_fbt,rx,zx,psi_fbt,psi_grid); hold on; plot(r0,z0,'ro');
title(s_fbt,'\psi_{PFC}'), xlabel(s_fbt,'R [m]'), ylabel(s_fbt,'Z [m]'); axis equal
caxis(s_fbt,[cax_min,cax_max]);
```

```
s_pc = subplot(1,3,2); % plasma current induced flux
contour(s_pc,rx,zx,psi_plasma,psi_grid); hold on; plot(r0,z0,'ro'); % subtract to obtain it
title(s_pc,'\psi_{plasma}'), xlabel(s_pc,'R [m]'), ylabel(s_pc,'Z [m]'); axis equal
caxis(s_pc,[cax_min,cax_max]);

s_tot = subplot(1,3,3); % total flux
contour(s_tot,rx,zx,psi_tot,psi_grid); hold on; plot(r0,z0,'ro');
title(s_tot,'\psi_{tot}'), xlabel(s_tot,'R [m]'), ylabel(s_tot,'Z [m]'); axis equal
caxis(s_tot,[cax_min,cax_max]);

% Setup colorbar
h = axes(fig,'visible','off'); c = colorbar();
caxis(h,[cax_min,cax_max]);
c.Location = 'southoutside';
c.Position(2) = 0.35*c.Position(2); c.Position(4) = 0.5*c.Position(4);
```



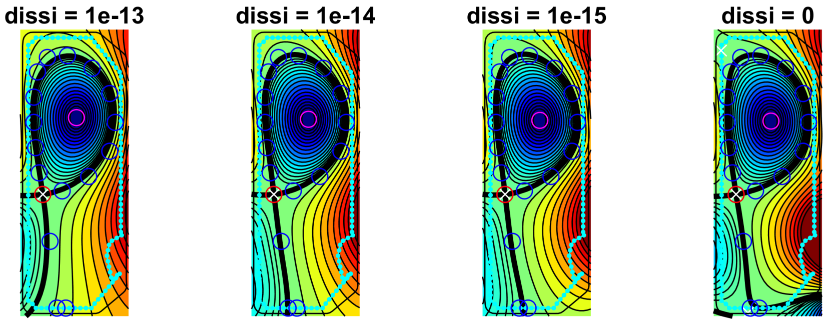

```
if print_figures == true
print_pdf(fig,'comparison_1d'); end
```

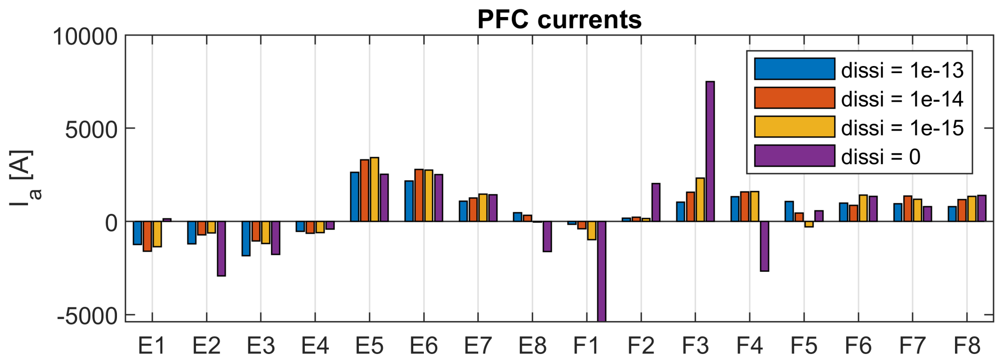
Exercise 2 - FBTE

Effect of power dissipation weight on the coil currents obtained from FBTE. The higer the weights the smaller the current but the worse the agreement of the solution with required LCFS. Look in particar at the bottom leg of the solution.

```
dissi = [1e-13,1e-14,1e-15,0];
```

```
exfbte(dissi,use_files);
```






```
if print_figures == true
    print_pdf(gcf,'fbte_dissi.pdf'); end
```

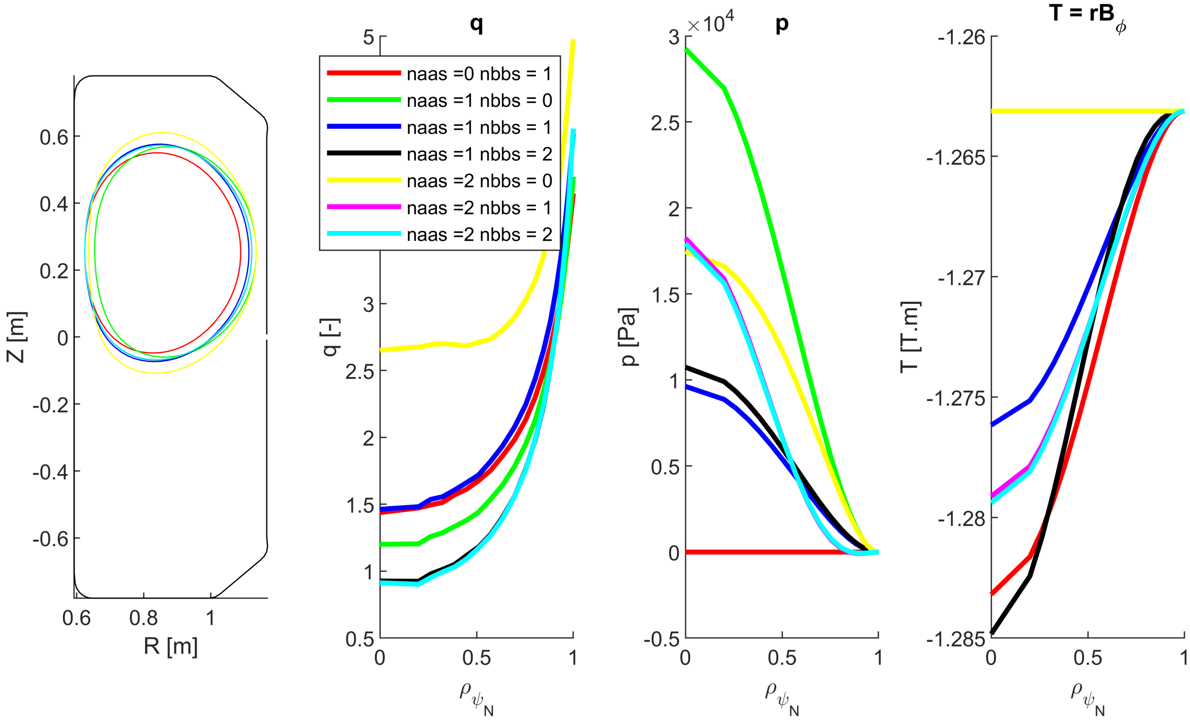
Exercise 3 - LIUQE Reconstruction

Study of the effect of the basis functions naas and nbbs to parametrize p' and TT' in the Grad-Shafranov equation. We see, that the equilibrium reconstruction is very sensitive to the choice of the basis functions used. To study this, we can use here the function exliuqe that scans for naas & nbbs between 0 and 2.

Limited configuration

```
shot = 40000;
time = 0.1;
exliuqe(shot, time, use_files);
```

```
Warning: TCV#40000 0.1000s/28: Multiple magnetic axes
Warning: TCV#40000 0.1000s/51: No convergence after 50 iterations - freezing plasma domain
Warning: TCV#40000 0.1000s/51: No convergence after 50 iterations - freezing plasma domain
```

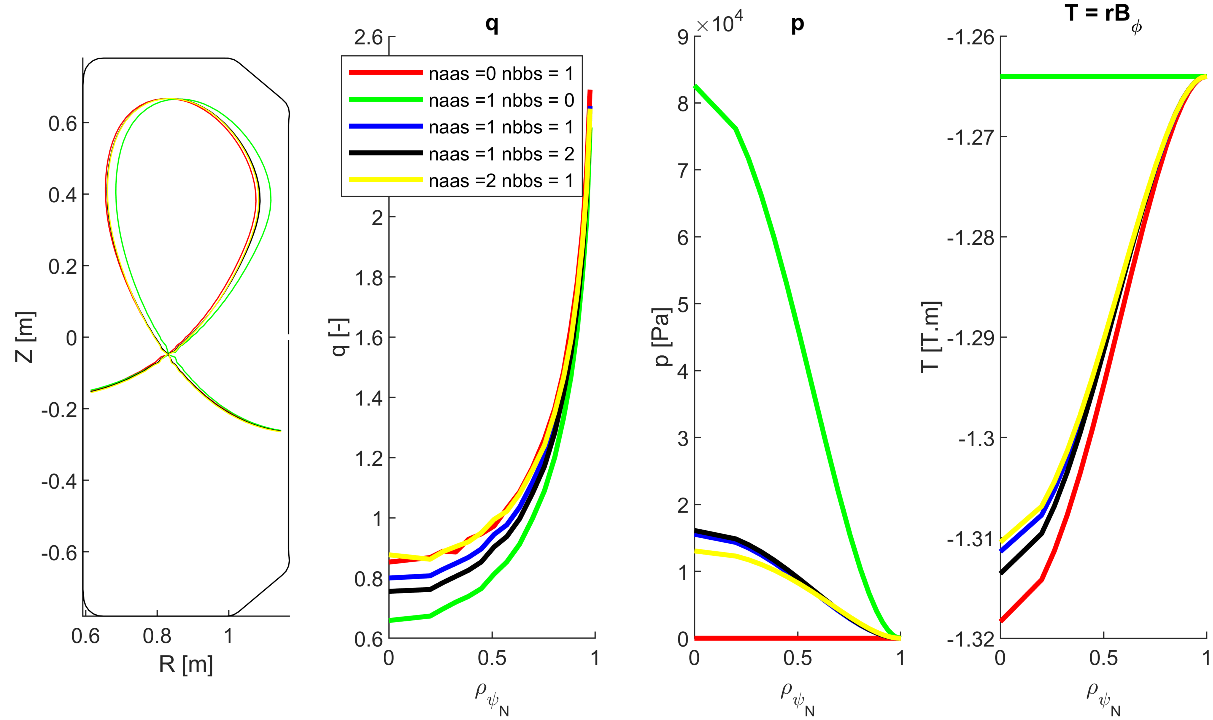



```
if print_figures == true
    print_pdf(gcf,'liuqe_limited'); end
```

Diverted configuration

```
shot = 40000;
time = 0.3;
exliuqe(shot, time, use_files);
```

```
Warning: TCV#40000 0.3000s/51: No convergence after 50 iterations - freezing plasma domain
Warning: TCV#40000 0.3000s/56: Multiple magnetic axes
Warning: TCV#40000 0.3000s/51: No convergence after 50 iterations - freezing plasma domain
Warning: TCV#40000 0.3000s/100: iterations exceeded
```




```
Warning: TCV#40000 0.3000s/9: Multiple magnetic axes
```

```
if print_figures == true
    print_pdf(gcf,'liuqe_diverted'); end
```

Additional functions for figure plots and pdf save

```
function print_pdf(fig, name)
    set(fig,'Units','Inches'); pos = get(fig,'Position');
    set(fig,'PaperPositionMode','Auto','PaperUnits','Inches',...
        'PaperSize',[pos(3), pos(4)]);
    print(fig,name,'-dpdf','-r0');
end
```