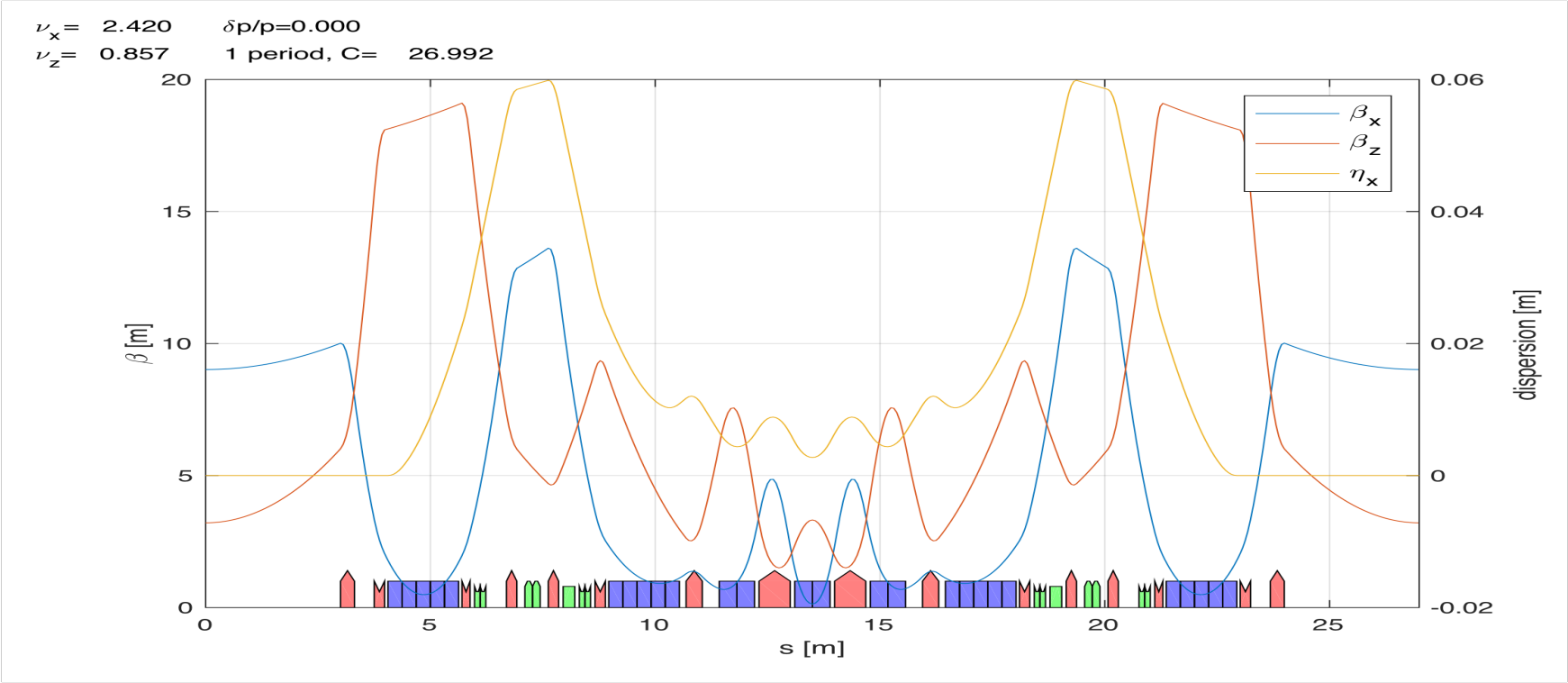
HEPS lattice and AT error and correction package

B. Nash

April 14, 2017

IHEP

HEPS lattice



Lattice creation

Drifts:

```
D1=atdrift('D1',3);
D2=atdrift('D2',0.44);
D3=atdrift('D3',0.075);
D4=atdrift('D4',0.075);
D5_1=atdrift('D5_1',0.1);
D5_2= atdrift('D5_2',0.46);
```

```
D6_1=atdrift('D6_1',0.18);
D7_1=atdrift('D7_1',0.1);
D8=atdrift('D8',0.075);
D9=atdrift('D9',0.15);
D10=atdrift('D10',0.376);
D11=atdrift('D11',0.1);
D12=atdrift('D12',0.1);
```

Bending magnets

```
BS11=atsbend('BS11',0.3142,0.00793448,0,'BndMPoleSymplectic4Pass','EntranceAngle',0.0109557,'ExitAngle',-0.00302124);...
BS12=atsbend('BS12',0.3142,0.00453399,0,'BndMPoleSymplectic4Pass','EntranceAngle',0.00302124,'ExitAngle',0.00151275);...
BS13=atsbend('BS13',0.3142,0.00377832,0,'BndMPoleSymplectic4Pass','EntranceAngle',-0.00151275,'ExitAngle',0.00529107);...
BS14=atsbend('BS14',0.3142,0.00302266,0,'BndMPoleSymplectic4Pass','EntranceAngle',-0.00529107,'ExitAngle',0.00831373);...
BS15=atsbend('BS15',0.3142,0.00264199,0,'BndMPoleSymplectic4Pass','EntranceAngle',-0.00831373,'ExitAngle',0.0109557);...

RBS15=atsbend('RBS15',0.3142,0.00264199,0,'BndMPoleSymplectic4Pass','EntranceAngle',0.0109557,'ExitAngle',-0.00831373);
RBS14=atsbend('RBS14',0.3142,0.00302266,0,'BndMPoleSymplectic4Pass','EntranceAngle',0.00831373,'ExitAngle',-0.00529107);
RBS13=atsbend('RBS13',0.3142,0.00377832,0,'BndMPoleSymplectic4Pass','EntranceAngle',0.00529107,'ExitAngle',-0.00151275);
RBS12=atsbend('RBS12',0.3142,0.00453399,0,'BndMPoleSymplectic4Pass','EntranceAngle',0.00151275,'ExitAngle',0.00302124);
RBS11=atsbend('RBS11',0.3142,0.00793448,0,'BndMPoleSymplectic4Pass','EntranceAngle',-0.00302124,'ExitAngle',0.0109557);

BQ1=atsbend('BQ1',0.393,0.00614659,-2.24461,'BndMPoleSymplectic4Pass','EntranceAngle',0.00614659);
BQ2=atsbend('BQ2',0.393,0.00614659,-2.24461,'BndMPoleSymplectic4Pass','ExitAngle',0.00614659);

BQC1=atsbend('BQC1',0.393,0.00933379,-2.2,'BndMPoleSymplectic4Pass','EntranceAngle',0.00933379);
```

Lattice creation cont…

Quadrupoles

```
Q1=atquadrupole('Q1',0.31,2.5246732,'StrMPoleSymplectic4Pass');
Q2=atquadrupole('Q2',0.23,-2.5136888,'StrMPoleSymplectic4Pass');
Q3=atquadrupole('Q3',0.18,-2.4980369,'StrMPoleSymplectic4Pass');
Q4=atquadrupole('Q4',0.23,2.3752118,'StrMPoleSymplectic4Pass');
Q5=atquadrupole('Q5',0.23,2.5487042,'StrMPoleSymplectic4Pass');
Q6=atquadrupole('Q6',0.23,-2.5965951,'StrMPoleSymplectic4Pass');
Q7=atquadrupole('Q7',0.36,3.9266853,'StrMPoleSymplectic4Pass');
Q8=atquadrupole('Q8',0.694,3.9891663,'StrMPoleSymplectic4Pass');
```

Sextupoles

```
SD2=atsextupole('SD2',0.125,-83);
SF=atsextupole('SF',0.17,85.780238);
OF=atsextupole('OF',0.26,0,'PolynomB',[0 0 0 -1950]);
SD1=atsextupole('SD1',0.125,-104.45627);
```

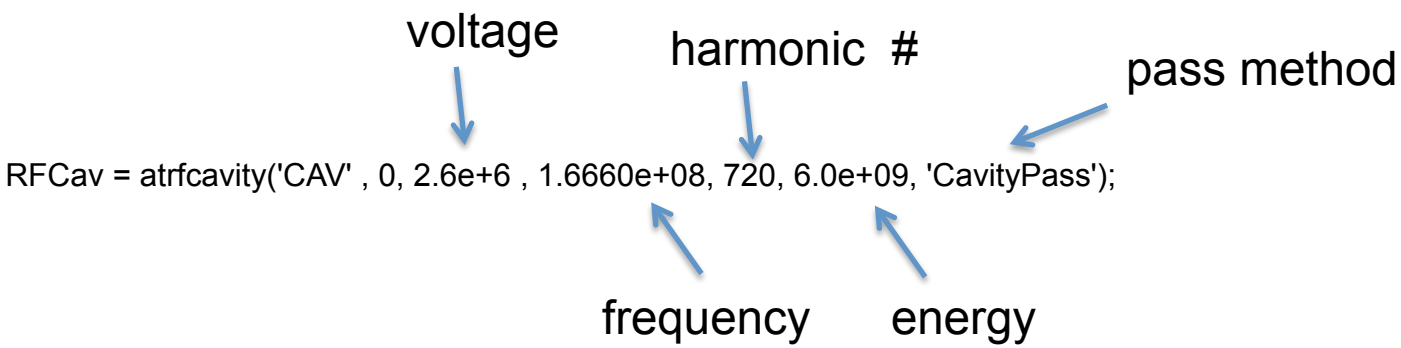
Put these together in order

```
cell = {D1;Q1;D2;Q2;D3;...
        BS11;BS12;BS13;BS14;BS15;...
        D4;Q3;D5_1;SD2;SD2;D5_2;...
        Q4;D6_1;SF;M1;SF;
        D6_1;Q5;D7_1;OF;D7_1;SD1;SD1;...
        D7_1;Q6;D8;RBS15;...
        RBS14;RBS13;RBS12;RBS11;...
        D9;Q7;D10;BQ1;BQ2;...
        D11;Q8;D12;BQC1;BQC1;
        D12;Q8;D11;BQ2;BQ1;D10;...
        Q7;D9;RBS11;RBS12;RBS13;RBS14;RBS15;...
       D8;Q6;D7_1;SD1;SD1;D7_1;OF;D7_1;Q5;...
       D6_1;SF;M1;SF;D6_1;Q4;D5_2;SD2;SD2;...
       D5_1;Q3;D4;BS15;BS14;BS13;BS12;BS11;...
       D3;Q2;D2;Q1;D1;...
};
```

Make 48 cells:

ring0 = repmat(cell,48,1);

RF Cavity



Add RF cavity to ring:

```
ring=[{RFCav};ring0];
```

Now add correctors and BPMs

```
CHV0=atcorrector('CHV0',0.1,[0,0]);

        CHV0.PassMethod='StrMPoleSymplectic4Pass';
        CHV0.PolynomB=[0 0];
        CHV0.PolynomA=[0 0];
        CHV0.MaxOrder=1;

CHV1=atcorrector('CHV1',0.1,[0,0]);

        CHV1.PassMethod='StrMPoleSymplectic4Pass';
        CHV1.PolynomB=[0 0];
        CHV1.PolynomA=[0 0];
        CHV1.MaxOrder=1;

BPM = atmonitor('BPM');
```

Now add misalignment errors in quadrupoles

```
ring = heps_Cor_BPM(1);

indq=find(atgetcells(ring,'Class','Quadrupole'));
```

Create random 1 micron errors:

```
dx=1e-6*randn(size(indq));
dy=1e-6*randn(size(indq));
```

Now set errors:

```
ringerr=atsetshift(ring,indq,dx,dy);
```

Visualize these errors

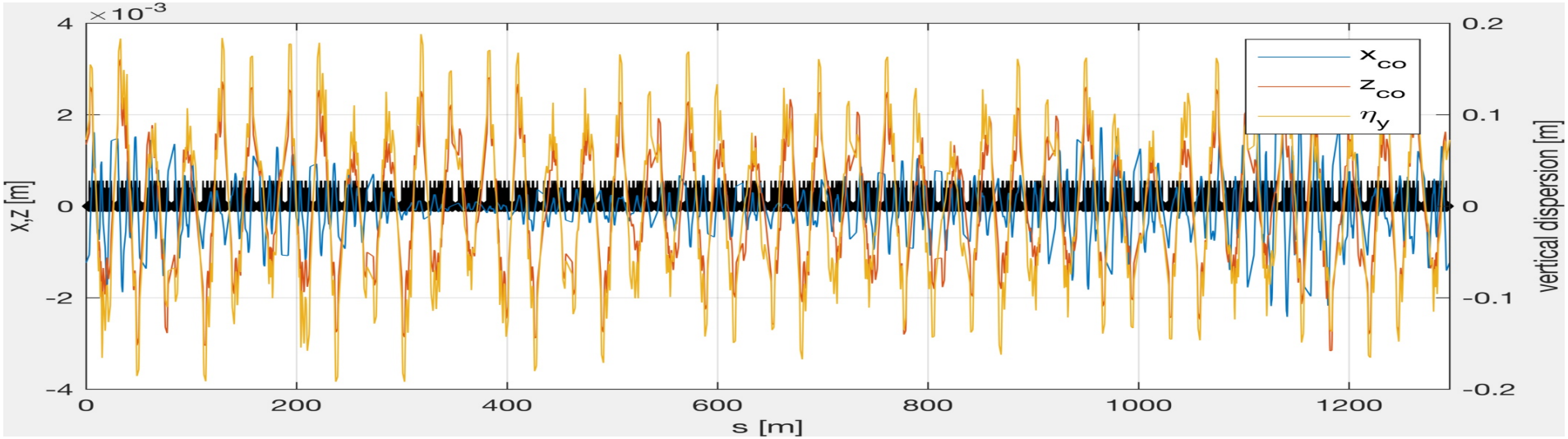
```
atplot(ring,'comment',[],@plClosedOrbit)
```

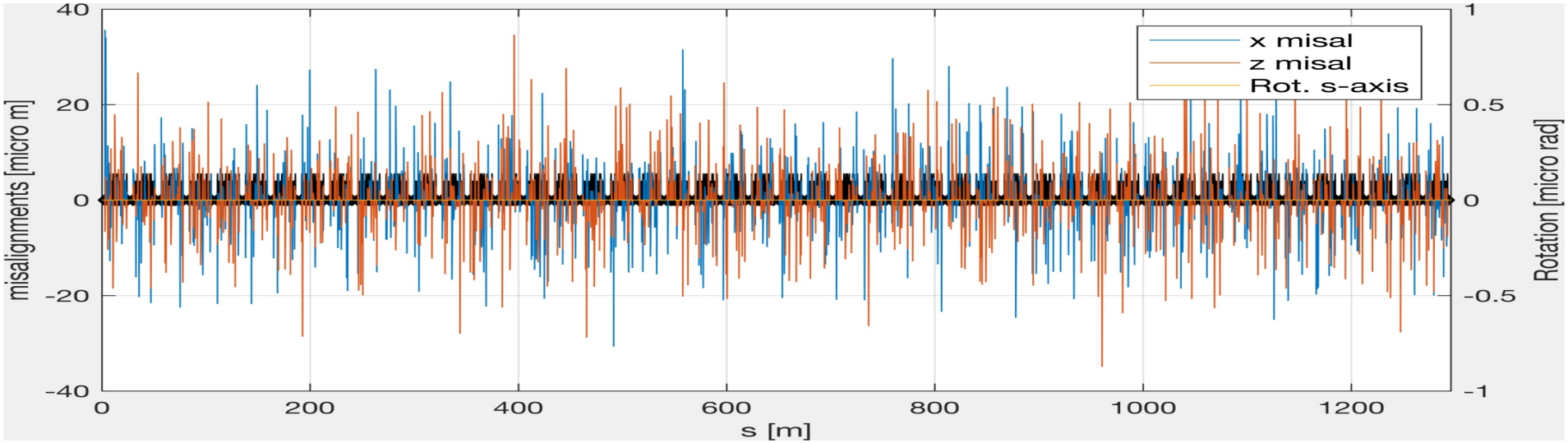
Closed Orbit plotting function

```
atplot(ringerr,[0,52],'comment',[],@pltmisalignments)
```

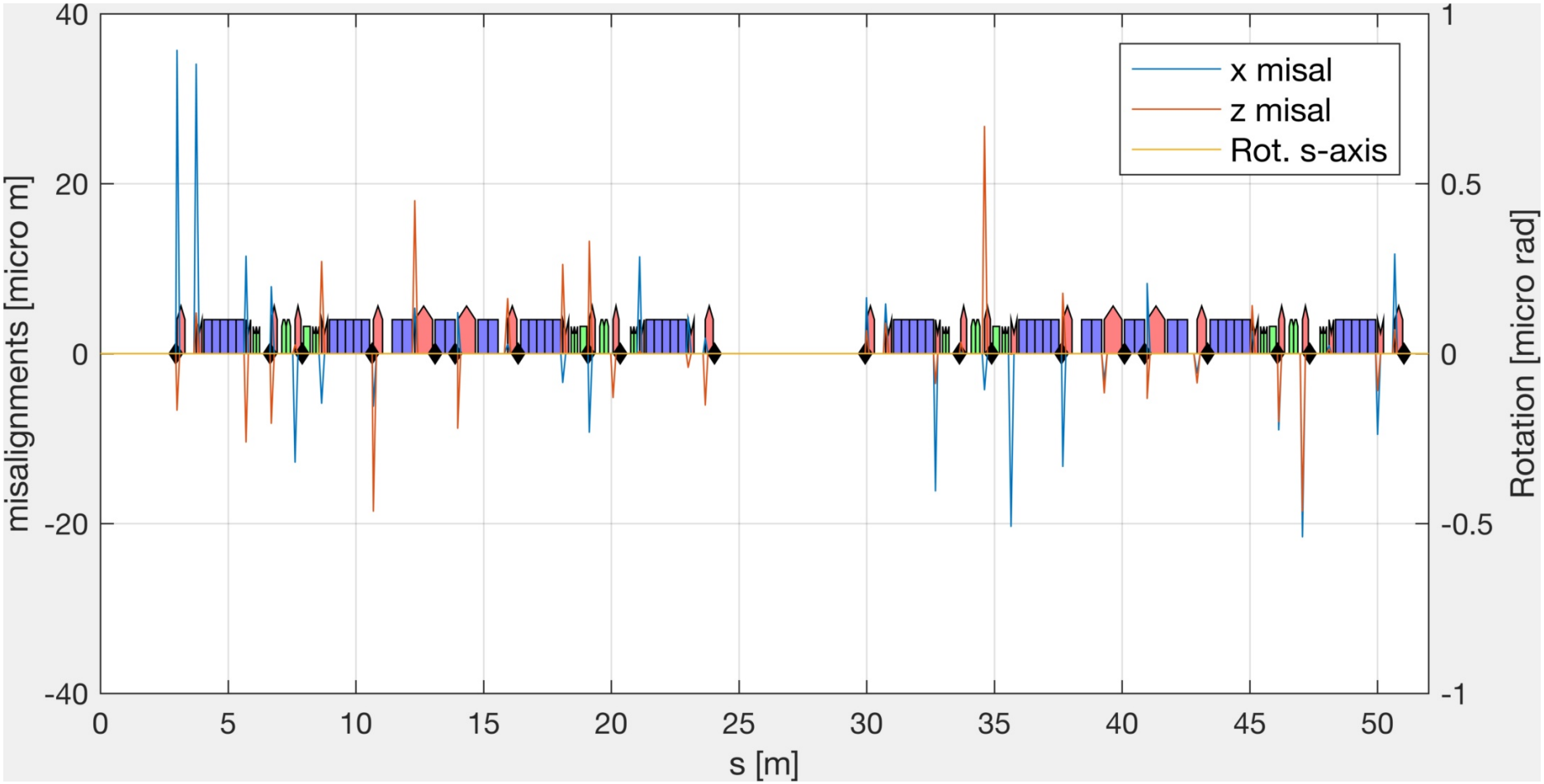
misalignment plotting function

viasualize misalignments and closed orbit distortion





zoom in on one cell



Now compute response matrix

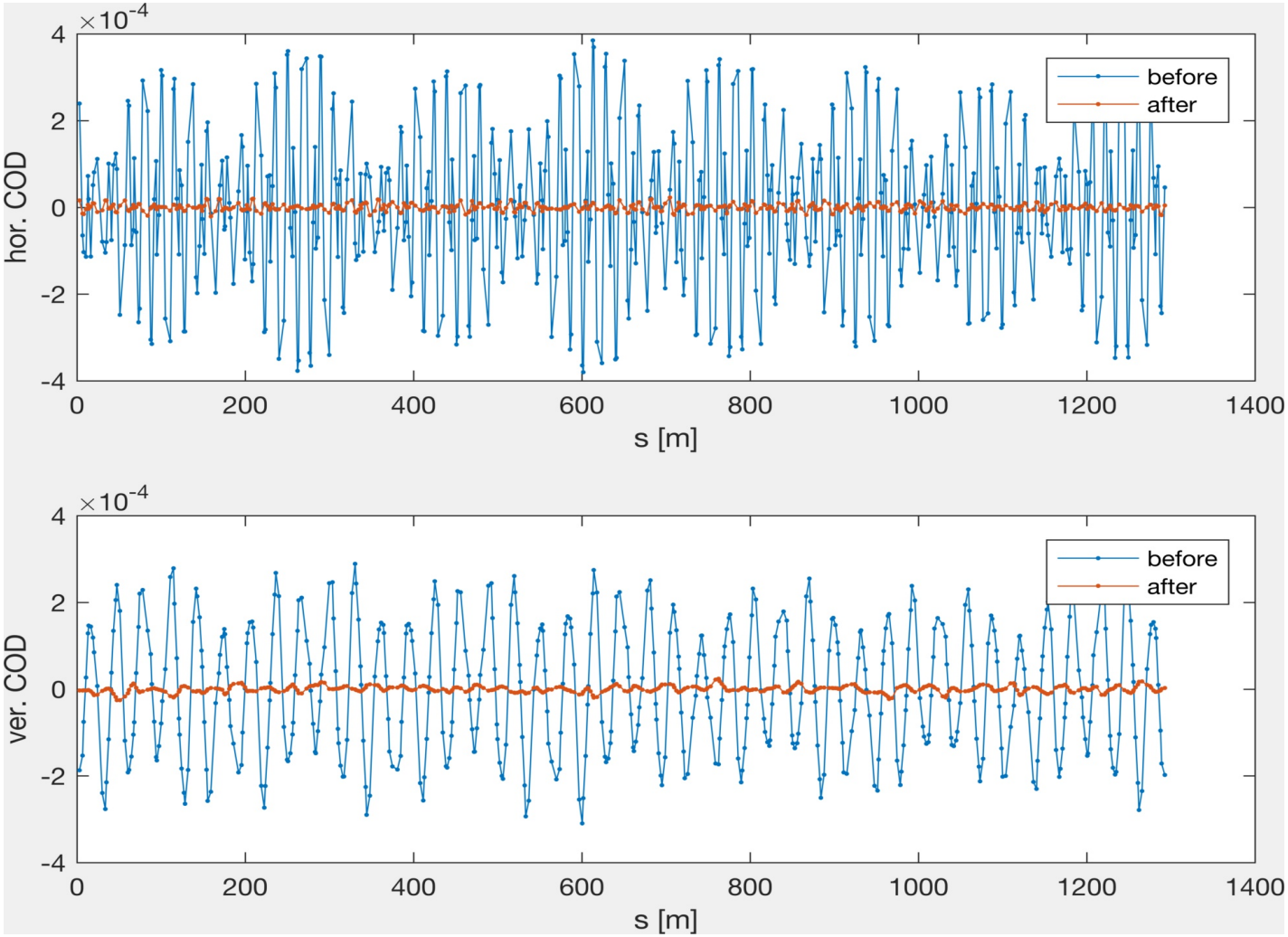
```
ModelRM...
    =getresponsematrices(...
    ring,indBPM,indHCor,...
    indVCor,indSCor,indQCor,...
    [],...
    [0 0 0 0 0 0]',...
    [1 2 3]);
```

```
ModelRM =

  struct with fields:

    OrbHCor: {[480×192 double]  [480×192 double]  [480×192 double]  [480×192 double]}
    OrbVCor: {[480×192 double]  [480×192 double]  [480×192 double]  [480×192 double]}
    OrbHDPP: [1×480 double]
    OrbVDPP: [1×480 double]
       kval: 1.0000e-04
      delta: 1.0000e-03
```

Now do orbit correction with error lattice and response matrix

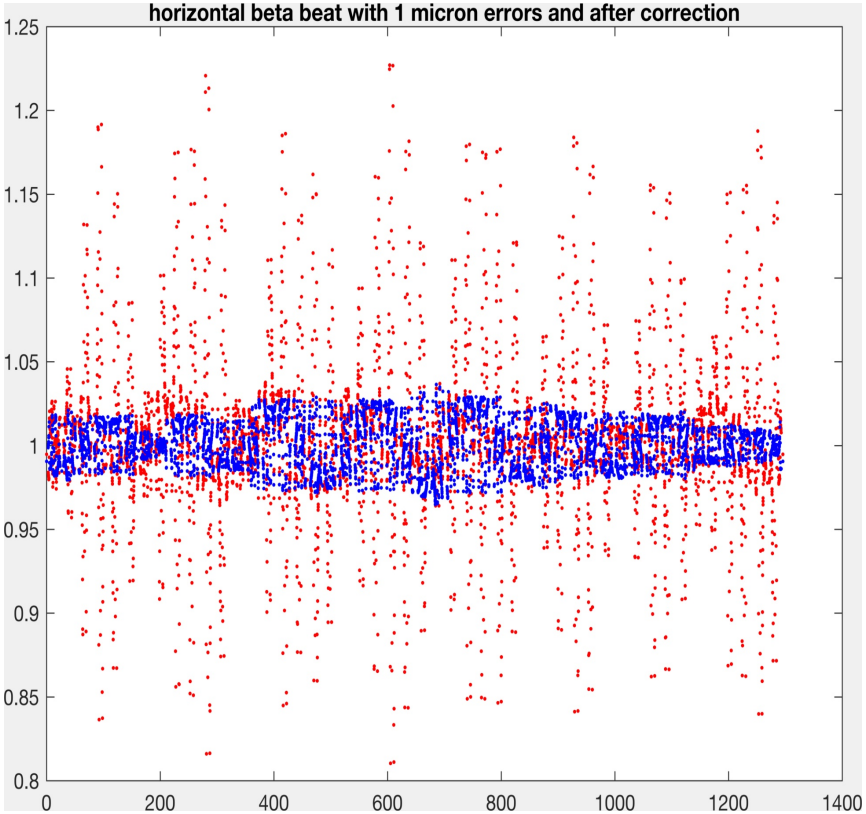
```
[rcor,inCOD,hs,vs]=atcorrectorbit(rerr,..
.
    indBPM,...          BPM indices
    indHCor',...        Horizontal corrector indices
    indVCor',...        Vertical corrector indices
initial closed orbit guess  [0 0 0 0 0 0]',...
    [50 50],...         2xNiter eigenvectors for correction H and V at each iteration
    [false true],...    correct [dpp mean0]
    1.0,...             scale factor
response matrix         ModelRM,...
    zeros(2,length(indBPM)),...
steererlimit            [],...
    true);              printouttext
```

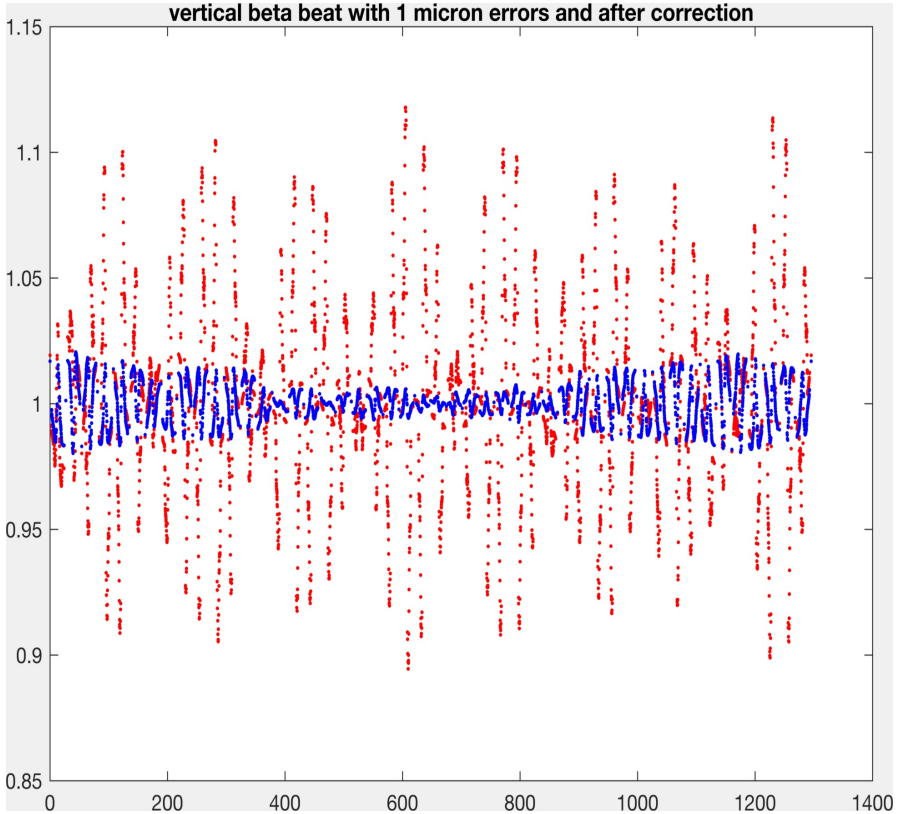
COD results for 1 micron misalignments



Orbit correction iter 1,
n-eig: 50, 50,
before --> after
oX: 109.785 -> 6.541um
oY: 68.739 -> 6.935um

beta beat results





Issues along the way

closed orbit not found issue

two times: once issue was need to supply a good guess, since dynamics were very unstable. Second time, issue was need to set RF frequency very precisely, (using atsetcavity), dependent on circumference

Needed to transpose corrector indices.
--Minor bug to fix in code

Response matrix code doesn't allow for CorrectorPass from atcorrector. Probably should be added as option.