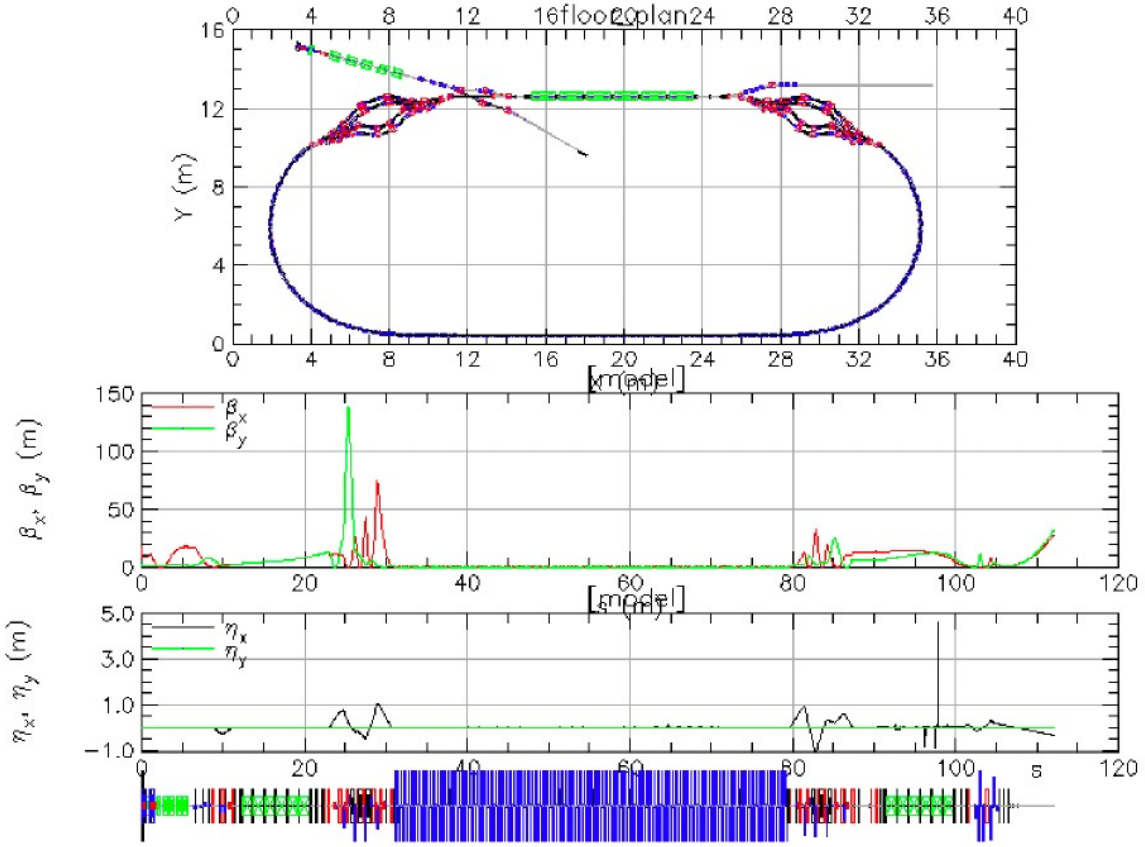
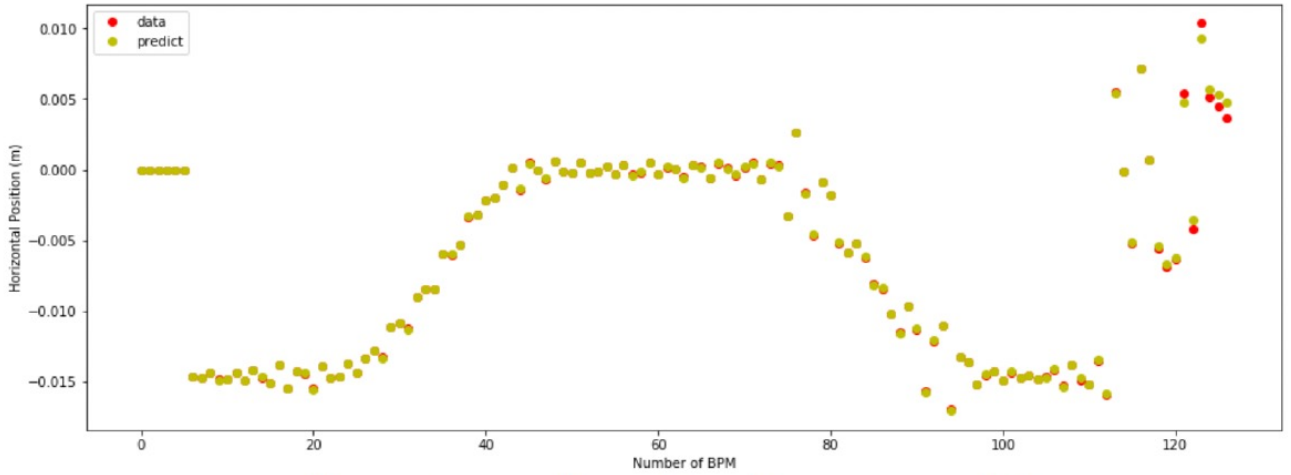

CLASSE
Cornell Laboratory for Accelerator-based ScienceS &Education
The Center for
BRIGHT
BEAMS
A National Science Foundation
Science & Technology Center
Brookhaven
National Laboratory
U.S. DEPARTMENT OF
ENERGY
Machine Learning & Bmad
Lucy Lin
Oct 6, 2022
Bmad Development Workshop
@BrookhavenLab

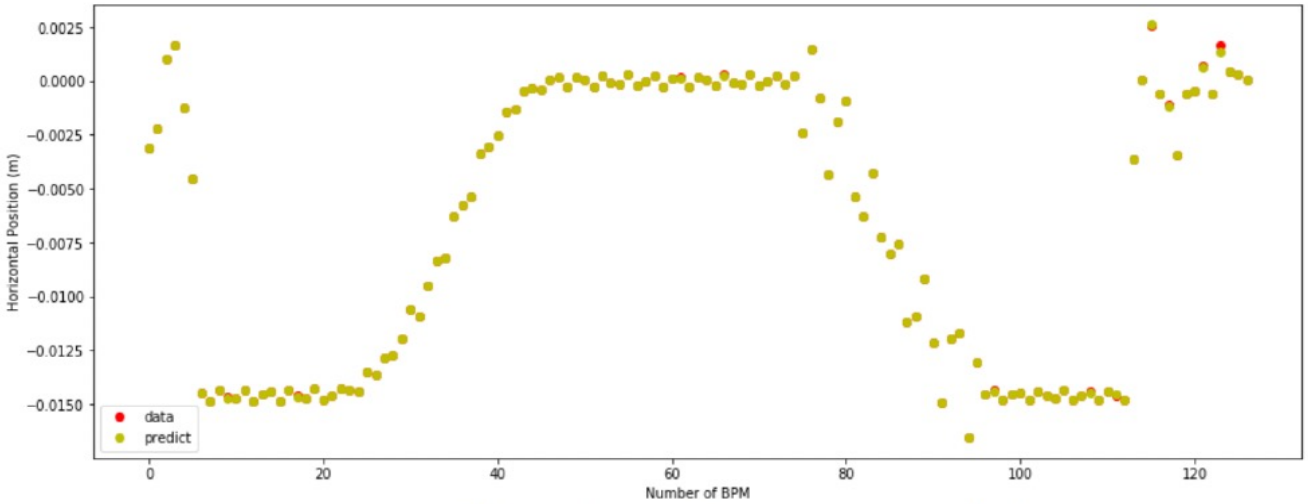
CBETA orbit prediction with NN

- Anyone can run pre-written lattice files to quickly get simulation data



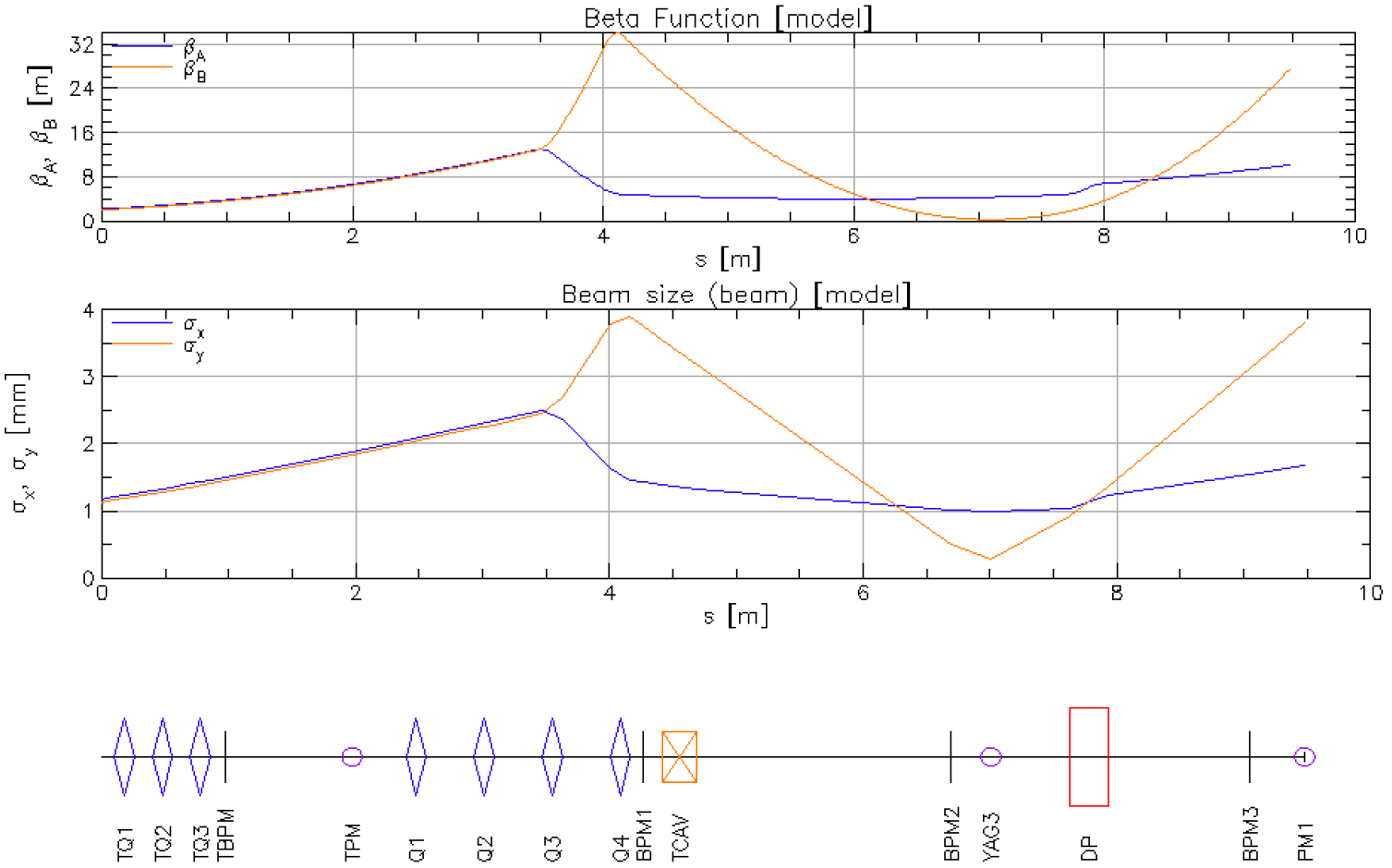


All 108 correctors – All 127 beam position monitors prediction



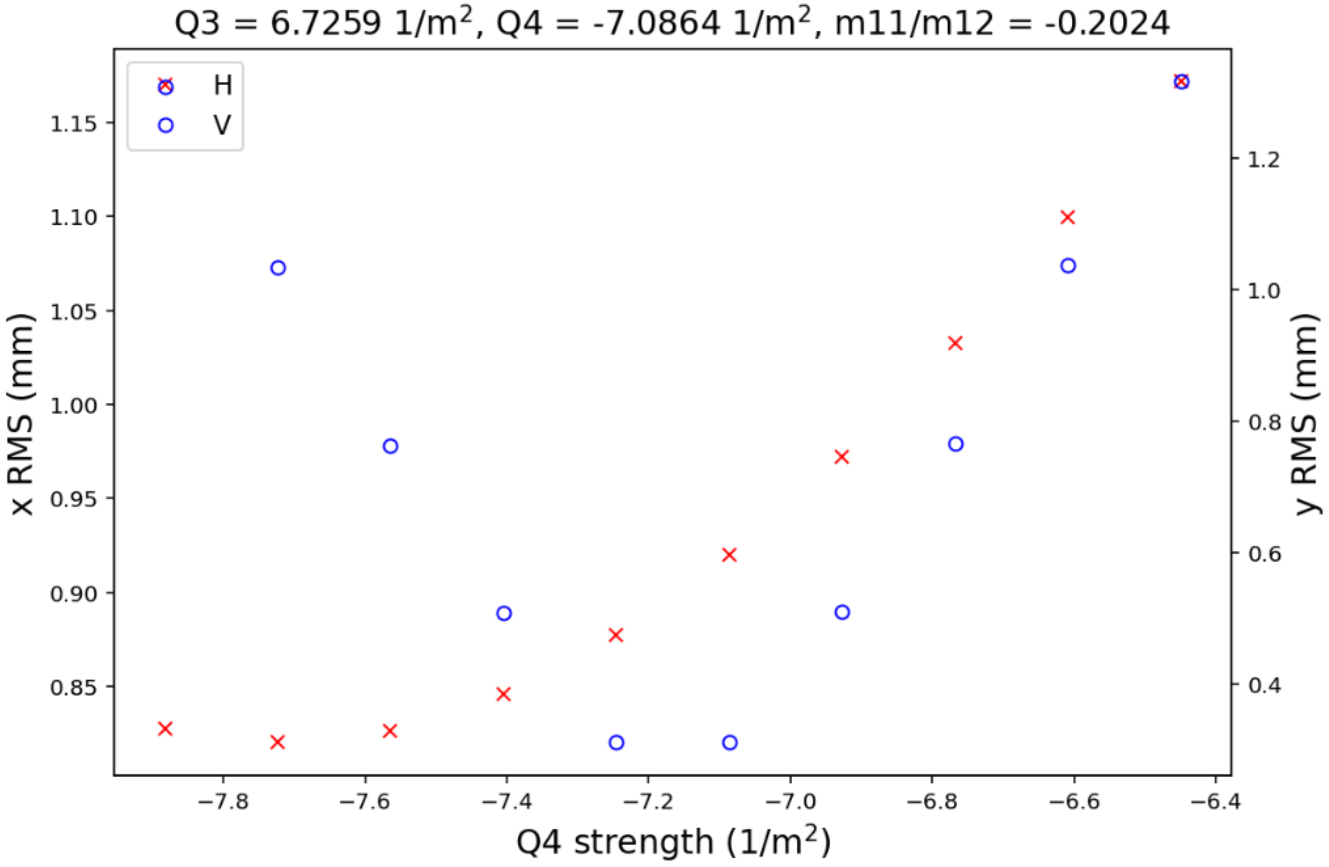
1 Cavity – All 127 beam position monitors prediction

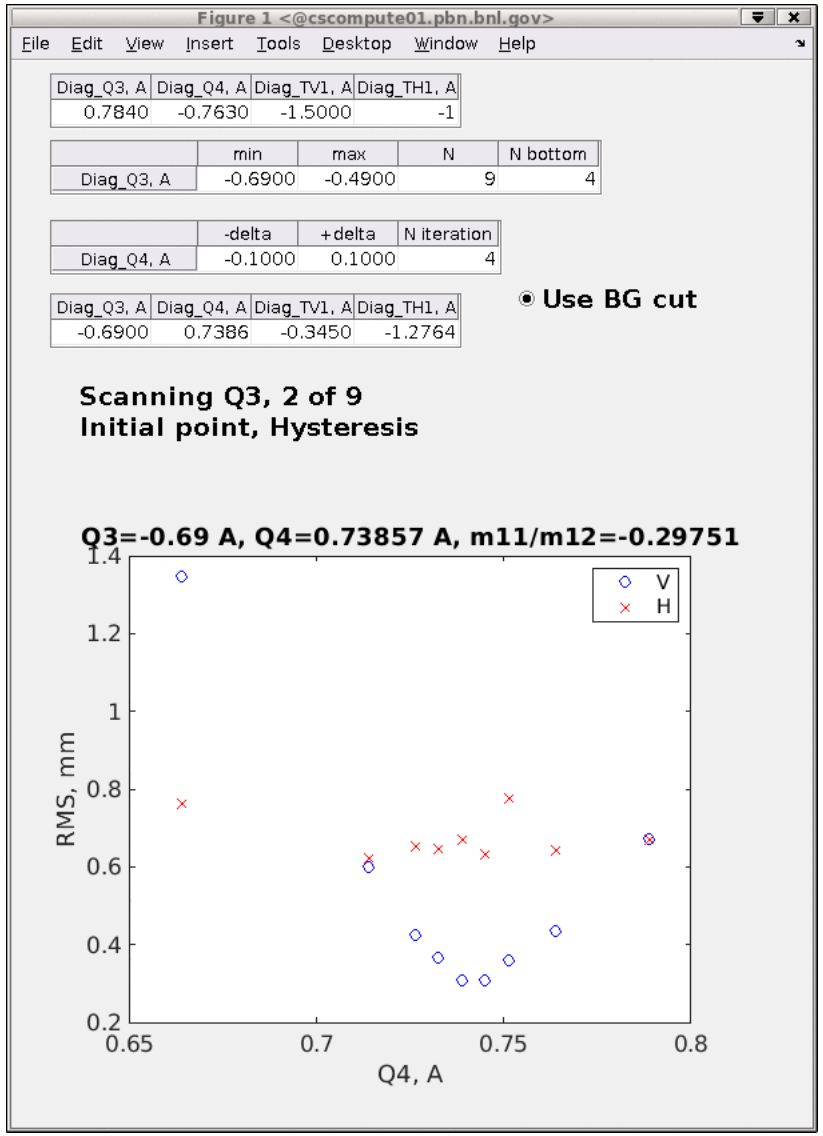
BMAD model building at BNL (CeC)



- Easy lattice creation for beamline
- Quickly obtain data for beam information (twiss function, beam size, emittance etc.)
- Conveniently generate simulated data to train machine learning algorithm

CeC emittance measurement in Bmad





BMAD model building at BNL (AGS)

- MAD-X to BMAD translation: much better interface with GUI, twiss function and orbit plots allow quick visual of lattice model (can be done very quickly with sequence files)

- Easy variable and datum definition → BMAD allows separate change of variables, many parameters defined with one variable in MAD-X

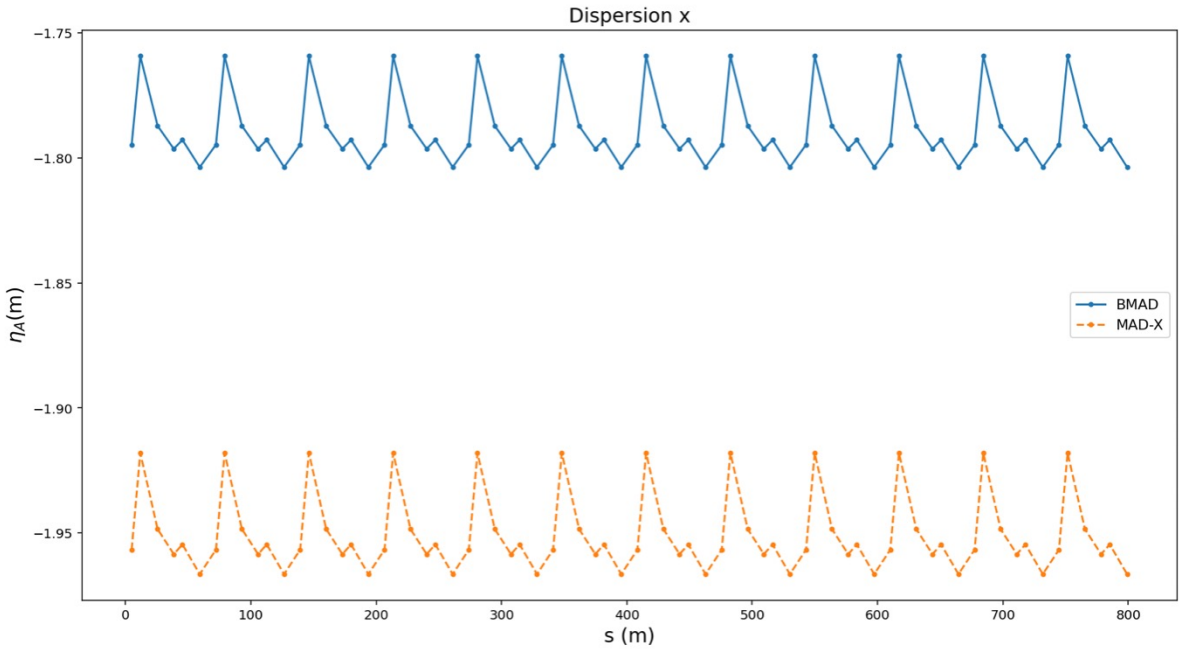
```
tao.cmd('show var quads.x')
```

```
['  Variable                Slave Parameters          Meas         Model        Design  Useit_opt',
 '  quads.x[1]              QH_F17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[2]              QH_G17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[3]              QH_H17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[4]              QH_I17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[5]              QH_J17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[6]              QH_K17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[7]              QH_L17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[8]              QH_A17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[9]              QH_B17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[10]             QH_C17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[11]             QH_D17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  quads.x[12]             QH_E17[K1]           0.0000E+00   -6.5349E-05   -6.5349E-05       T',
 '  Variable                Slave Parameters          Meas         Model        Design  Useit_opt']
```

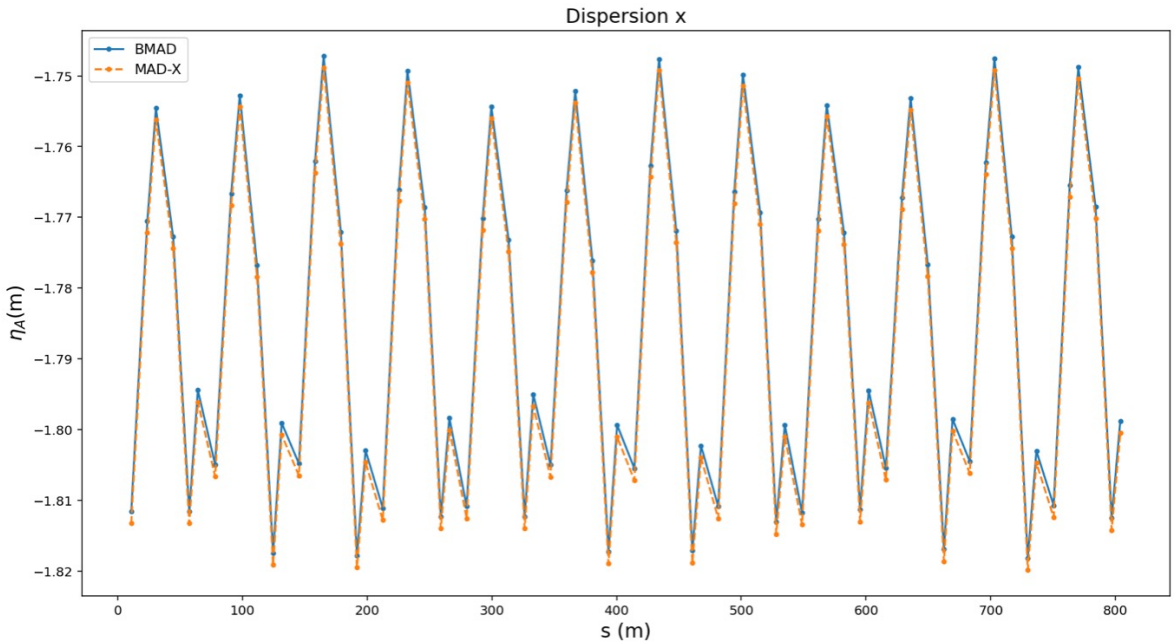
Note: difference in dispersion

MAD-X: $z = -c\Delta t(s), P_z = \frac{\Delta E}{E_0}$ ⟷ Agree only when $\gamma \gg 1$ ⟷ BMAD: $z = -\beta(s)c\Delta t(s), P_z = \frac{\Delta P}{P_0}$

$\gamma = 2.51$



$\gamma = 23.2$



BMAD and PyTao: best tool for ML

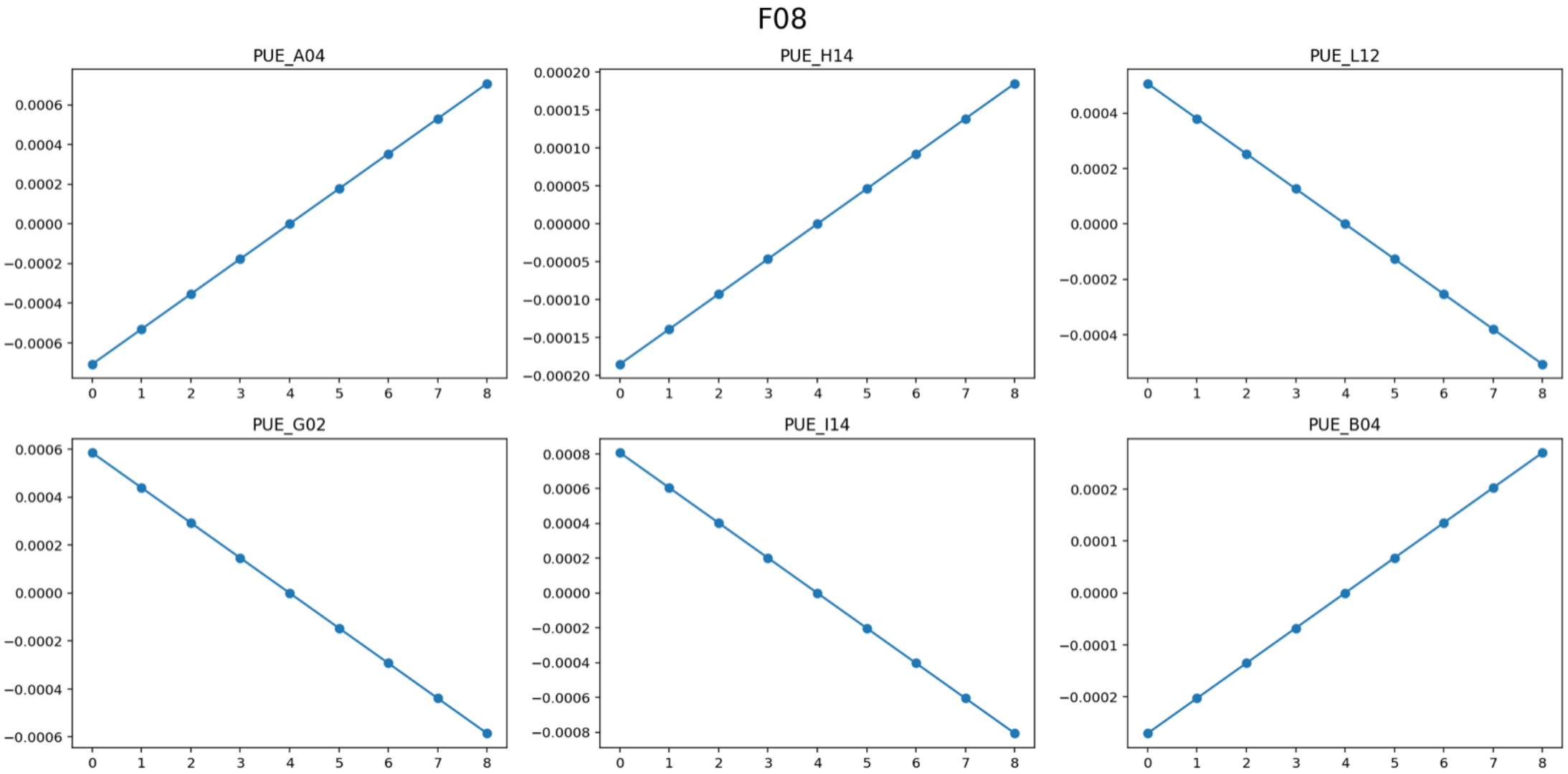
- Python interface: enable running simulations in Python scripts and Jupyter notebooks
- Get data with different control parameters in one go with for loops, without need to modify original lattice files
- Freedom to save data in any preferred form (i.e. combine into one huge data array and save in one file for easy fetch in the future)

BMAD and PyTao: best tool for ML

- Python command in Tao: output data in lists and arrays easy for python to handle

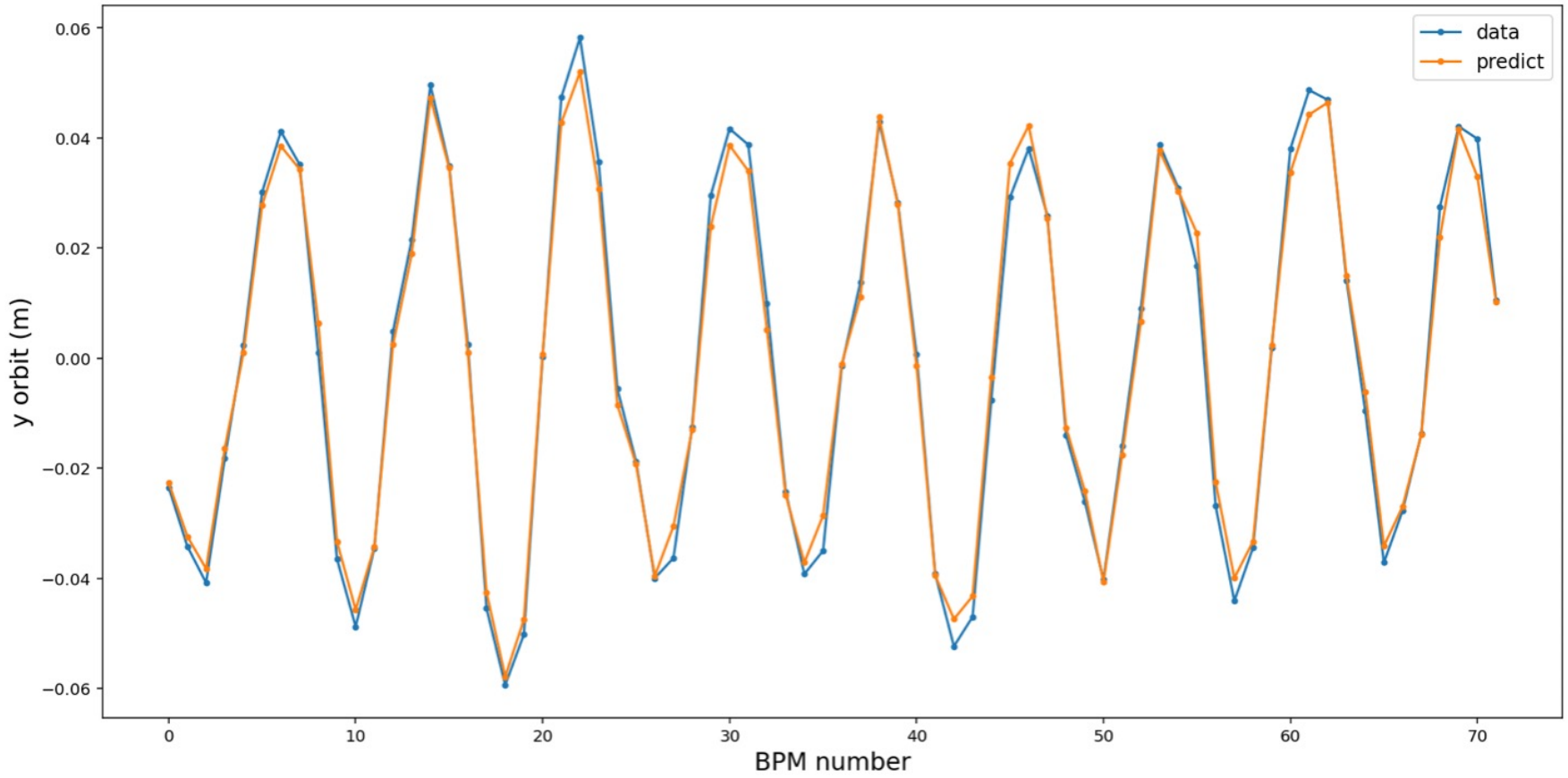
```
for i in range(1000):

    print('Generating random corrector settings...')

    # corrector settings with random values between +/- 20 Amp
    csets = np.random.uniform(-20,20,48) * k_cal

    print('Settings correctors...')
    for idx in range(1,49):
        setting = csets[idx-1]
        tao.cmd('set var correctors.y['+str(idx)+']|model = '+str(setting))

    print('reading bpm orbit data from tao ...')
    yll = tao.cmd('python data_d_array 1@orbit.bpm_y')
    yl = np.array([float(line.split()[2].replace(';','')) for line in yll])

    print(f'saving bpm orbit data to {datadir}...')

    fname = 'ormdata_'+str(i+1)

    dic = {'corr':csets, 'y':yl}
    pickle.dump(dic, open(datadir+fname+'.twiss', "wb"))

    print('zeroing all correctors ...')

    for idx in range(1,49):
        tao.cmd('set var correctors.y['+str(idx)+']|model = 0')

    print('Starting next data set...')
```

Orbit Response vs. One Corrector in Bmad



Train Neural Network for AGS ORM

Input 48 vertical corrector kick → Output 72 y orbit measured at BPM

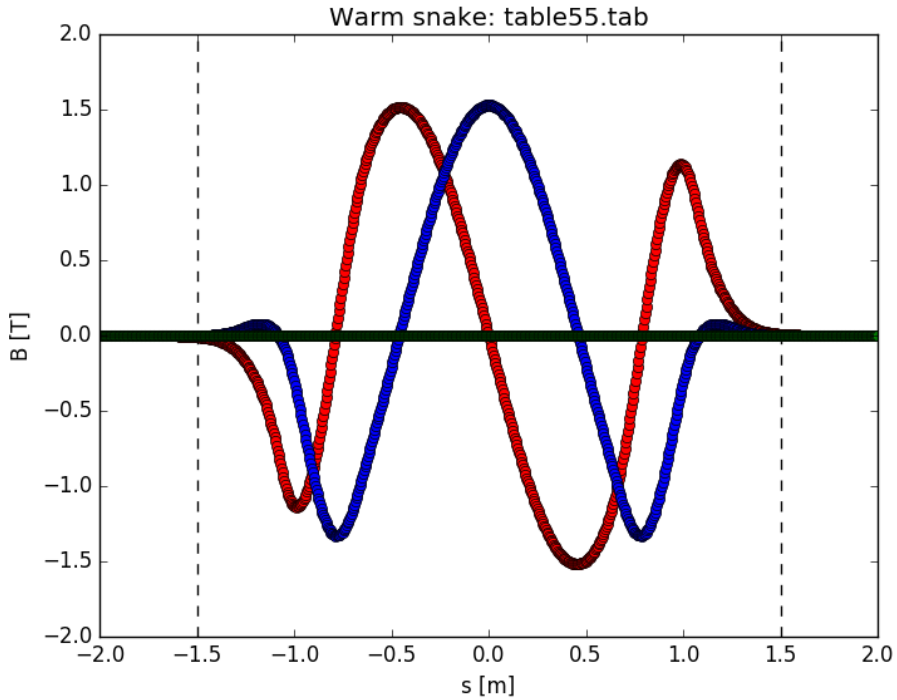


Ongoing: AGS Snakes

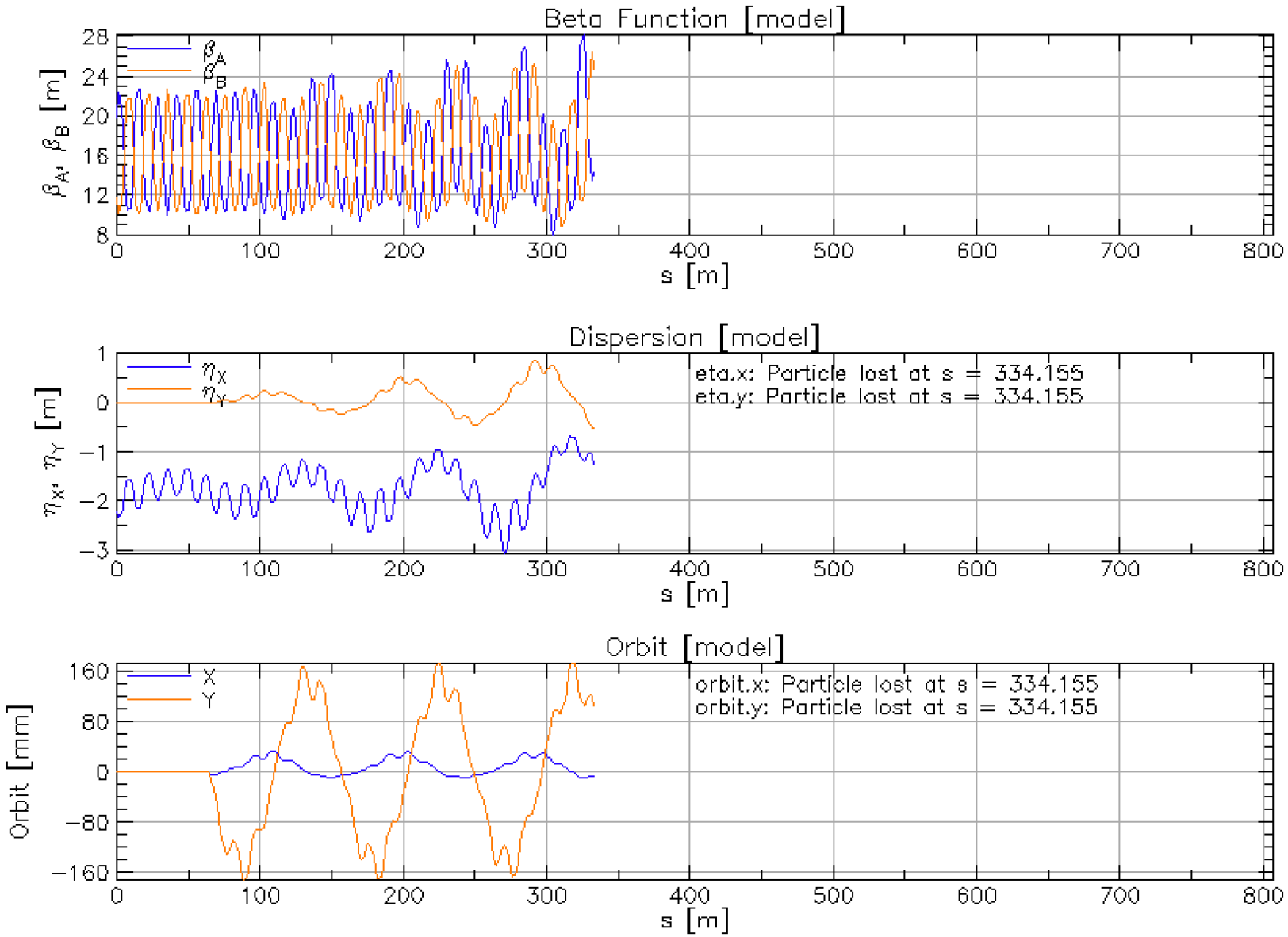
table55.bmad

```
{ geometry = xyz,
  field_type = mixed,
r0=(-0.0700, -0.0700, -2.0000),
dr=(5e-03, 5e-03, 5e-03),
pt(0, 0, 0) = (0, 0, 0, -1.28861E-04, -2.77401E-04, 6.65641E-05),
pt(0, 0, 1) = (0, 0, 0, -1.40711E-04, -2.82375E-04, 8.66296E-05),
pt(0, 0, 2) = (0, 0, 0, -1.52235E-04, -2.87364E-04, 1.06609E-04),
pt(0, 0, 3) = (0, 0, 0, -1.63426E-04, -2.92370E-04, 1.26501E-04),
pt(0, 0, 4) = (0, 0, 0, -1.74279E-04, -2.97392E-04, 1.46303E-04),
pt(0, 0, 5) = (0, 0, 0, -1.84786E-04, -3.02430E-04, 1.66014E-04),
pt(0, 0, 6) = (0, 0, 0, -1.94941E-04, -3.07485E-04, 1.85632E-04),
pt(0, 0, 7) = (0, 0, 0, -2.04739E-04, -3.12555E-04, 2.05156E-04),
pt(0, 0, 8) = (0, 0, 0, -2.14172E-04, -3.17642E-04, 2.24583E-04),
pt(0, 0, 9) = (0, 0, 0, -2.23233E-04, -3.22745E-04, 2.43913E-04),
pt(0, 0, 10) = (0, 0, 0, -2.34564E-04, -3.28099E-04, 2.56370E-04),
pt(0, 0, 11) = (0, 0, 0, -2.48158E-04, -3.33705E-04, 2.61951E-04),
pt(0, 0, 12) = (0, 0, 0, -2.61360E-04, -3.39327E-04, 2.67430E-04),
pt(0, 0, 13) = (0, 0, 0, -2.74163E-04, -3.44962E-04, 2.72804E-04),
pt(0, 0, 14) = (0, 0, 0, -2.86561E-04, -3.50612E-04, 2.78071E-04),
pt(0, 0, 15) = (0, 0, 0, -2.98546E-04, -3.56276E-04, 2.83230E-04),
pt(0, 0, 16) = (0, 0, 0, -3.10110E-04, -3.61952E-04, 2.88279E-04),
pt(0, 0, 17) = (0, 0, 0, -3.21247E-04, -3.67641E-04, 2.93216E-04),
pt(0, 0, 18) = (0, 0, 0, -3.31950E-04, -3.73341E-04, 2.98039E-04),
pt(0, 0, 19) = (0, 0, 0, -3.42211E-04, -3.79052E-04, 3.02747E-04),
pt(0, 0, 20) = (0, 0, 0, -3.59295E-04, -3.85397E-04, 3.11110E-04),
pt(0, 0, 21) = (0, 0, 0, -3.75923E-04, -3.91751E-04, 3.19353E-04),
pt(0, 0, 22) = (0, 0, 0, -3.92087E-04, -3.98111E-04, 3.27476E-04),
pt(0, 0, 23) = (0, 0, 0, -4.07779E-04, -4.04477E-04, 3.35478E-04),
pt(0, 0, 24) = (0, 0, 0, -4.22993E-04, -4.10846E-04, 3.43356E-04),
pt(0, 0, 25) = (0, 0, 0, -4.37721E-04, -4.17217E-04, 3.51109E-04),
pt(0, 0, 26) = (0, 0, 0, -4.51956E-04, -4.23588E-04, 3.58735E-04),
pt(0, 0, 27) = (0, 0, 0, -4.65690E-04, -4.29956E-04, 3.66233E-04),
pt(0, 0, 28) = (0, 0, 0, -4.78916E-04, -4.36320E-04, 3.73603E-04),
pt(0, 0, 29) = (0, 0, 0, -4.96497E-04, -4.43020E-04, 3.83303E-04),
-:**-  table55.bmad   Top L2     (Fundamental)
```

- Helix dipole + solenoid
- Before: use matrix file with closest gamma values → unphysical jumps in between
- Future: plan to use fieldmap files like in Zgoubi

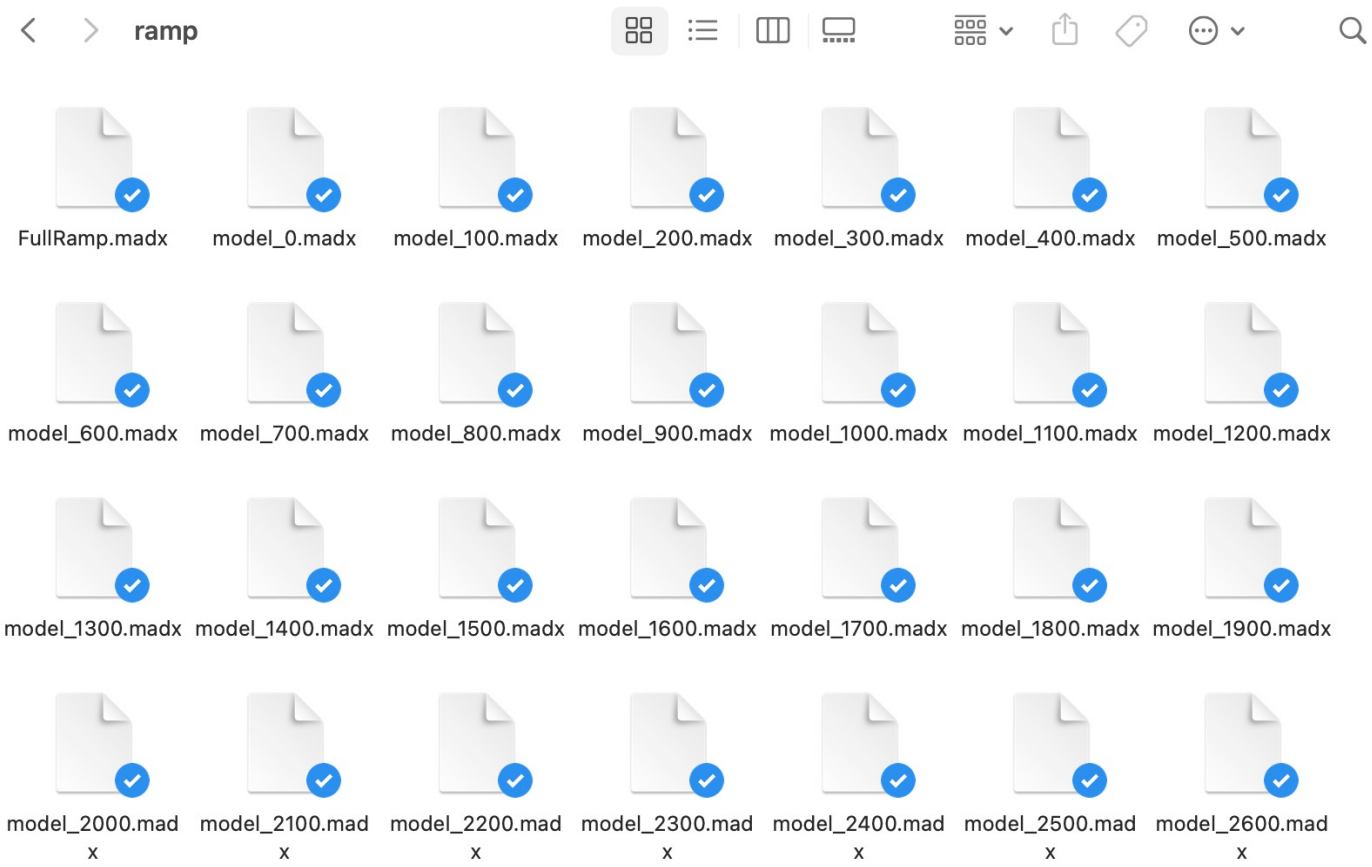


Add snake fieldmaps: need optics tuning



Ongoing: AGS ramping

MAD-X: cmd line tool ‘Snapramp’ to produce lattice files and twiss results at different time stamps



- Creating script to do ramping in Bmad
- Complication: magnet kicks change with energy → Bmad doesn’t recalculate predefined polynomials
- Either: make ramper element for every magnet
- Or: create time stamp files and interpolate like MAD-X


CLASSE
Cornell Laboratory for Accelerator-based ScienceS &Education
The Center for
BRIGHT
BEAMS
A National Science Foundation
Science & Technology Center
Brookhaven
National Laboratory
U.S. DEPARTMENT OF
ENERGY
Thanks!
@BrookhavenLab