EIC Tracking Simulations with Bmad

Matt Signorelli

Cornell ERL/EIC Group

PI: Georg Hoffstaetter

Cornell Laboratory for Accelerator-based Sciences and Education (CLASSE)

BROOKHAVEN NATIONAL LABORATORY

a passion for discovery

Office of Science U.S. DEPARTMENT OF ENERGY

Work Overview

EIC ESR Polarization

- Linear/nonlinear spin tracking with radiation
  - Does polarization hold up well with nonlinearities?
  - Which exact element(s) and/or nonlinear effect causes disagreement?

- Cross-checking results between various modern codes

- Verifying polarization robustness (i.e. with ϵ_y creator)

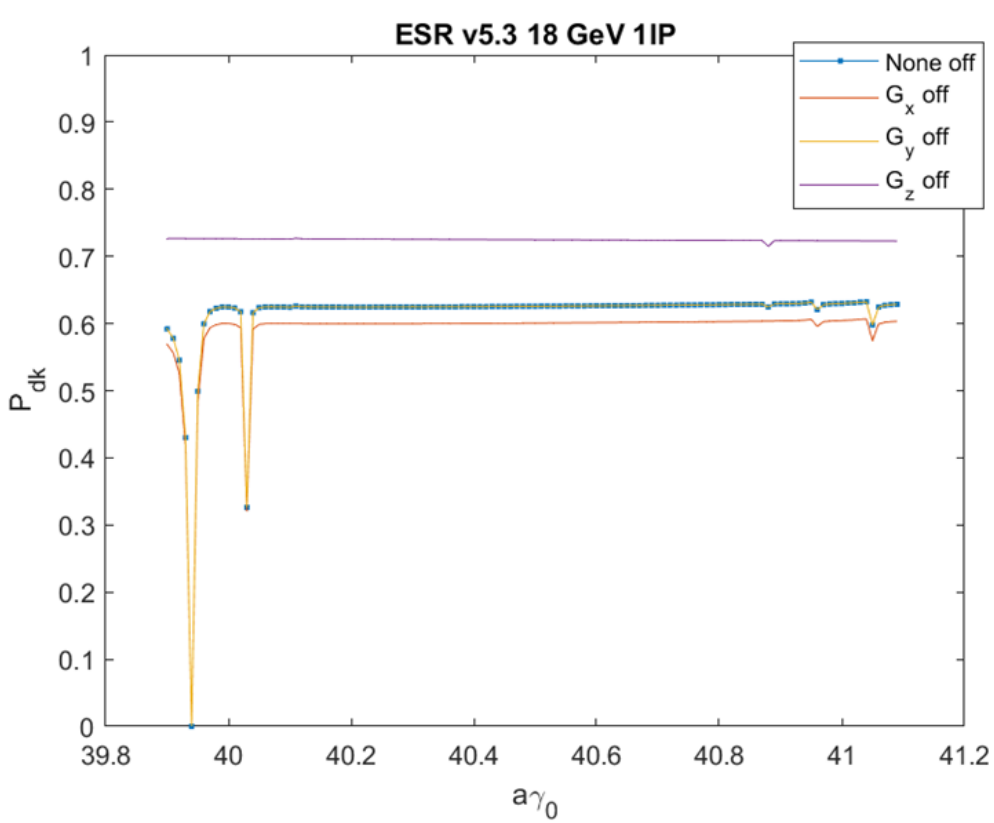
  Primarily use **Tao** and **Long Term Tracking** in my work

Workflow with Bmad

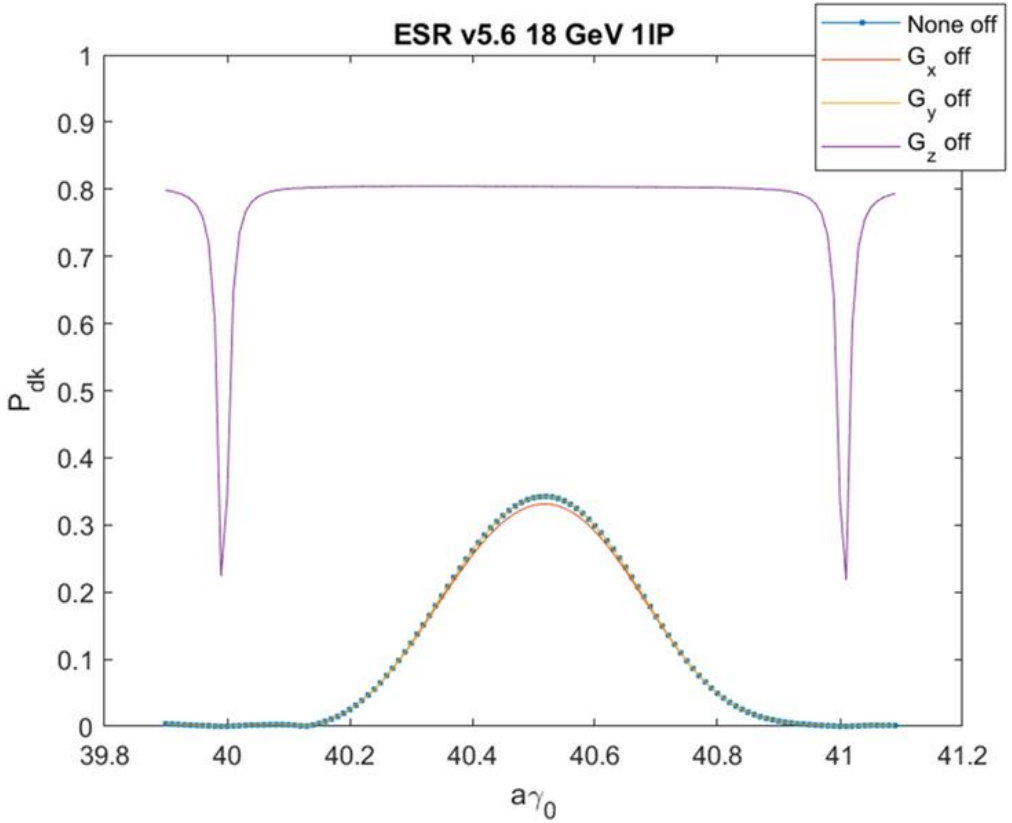
- **First step: converting a MAD-X lattice to Bmad**
  - ✓ Conversion scripts all written and generally work as intended
  - × Hidden in util_program directory, not widely accessible/simple for new Bmad-er/current MAD-X-er to use (barrier to new user)

- **Checking analytical polarizations/polarization times**
  - ✓ All 1-turn calculations, w/ and w/o individual modes (Tao)
  - ✓ G-matrices, quarternion maps, spin direction (Tao)
  - ✓ Energy scans of polarization (pyTao: polarization_vs_energy.py)
  - ✓ Fast first order spin propagation *Sprint* (Tao)

Effects of Longitudinal Spin Match



$G_z = 0$



$G_z \neq 0$

**nonlinearities give much lower actual P_{dk}*

- × Hidden in separate Python script, not widely accessible/simple for new Bmad-er/current MAD-X-er to use (barrier)

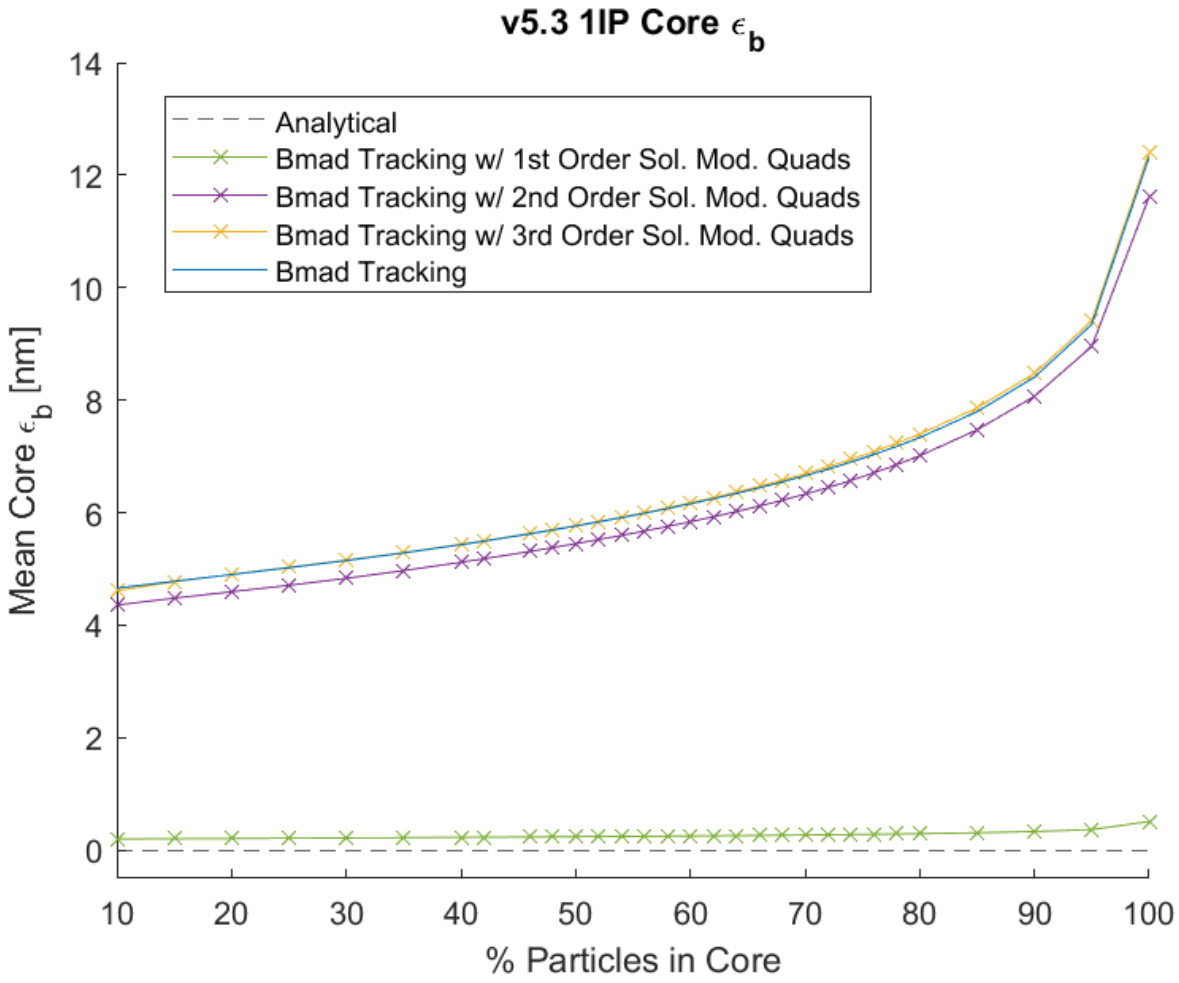
Workflow with Bmad

- **Spin tracking with radiation (Long Term Tracking)**
  - ✓ Parallelization built in
  - ✓ Most accurate 6D radiation kick
  - ✓ Simple sawtooth orbit correcting (taper in Tao)
  - ✓ Many different methods (map tracking, PTC, Bmad tracking, …)
  - ✓ Output (turn-by-turn, individual particle coords., core emittances, …)
  - ✓ User-specified order, even through individual elements
  - ✓ …

Results – v5.3

Core Emittance

- Almost entire effect presents with 2nd order quadrupoles in coupled regions
- **Coupling in solenoid modules appears to not cancel for off-energy particles, creating ~5 nm of vertical emittance**
- **Polarization?**



Workflow with Bmad

- **ϵ_y-creator: inserting a vertical chicane, spin matching (Tao)**
  - ✓ Fast first order spin matching *Sprint*
  - ✓ Several useful built in optimizers and adjustable parameters
  - ✓ Combine consecutive elements
  - × Setting up optimization:

```
show lat m1 -attribute beta_a@f20.17 -attribute beta_b@f20.17
show lat m1 -attribute alpha_a@f20.17 -attribute alpha_b@f20.17
show lat m1 -attribute phi_a@f20.17 -attribute phi_b@f20.17
```

copy + paste then a bunch of set up of datum file

→ write tao? (match optics to element, match G-matrices 1-turn, etc.)

Bmad Restructuring Ideas

- **Barriers to new users, MAD-X-ers must be removed**
  - Lattice conversions hidden in Python script
  - Installing entire repository + compiling lengthy, involved
    - MAD-X is a single executable
  - Locations of separate programs not in Tao, compiling and using them
    - e.g. Need to export environment variable in .bashrc for MPI version of Long Term Tracking
  - Fortran is scary to people
  - New Bmad programs need to be in Fortran? PyTao?
    - Need to fully understand entire Bmad structure to write new programs?

Bmad Restructuring Ideas

- **We need:**
  - Easy and quick to get started for the new user
  - Easily able to add and use all of Bmad's programs as needed

- **Propose a single, unified program**
  - One program for users to download
  - By default includes the basic functions (Tao, lattice conversions, …)
  - Simple way to download specific functions/add-ons as needed
  - Open file exchange where any user can upload/download custom functions or scripts to their Bmad environment
  - If possible, written in Julia

Bmad Restructuring Ideas

Just a rough idea...

```
Bmad > translate, from="madx", to="bmad", file="esr-main.madx"
Bmad > call, file="esr-main.bmad"

Using line "ring"

Bmad > show, lat
...
Bmad > call, file="fodo.bmad", uni=2
Bmad > set, uni=2
Bmad > show, emit
...
Bmad > download, function="depol_scan"
Bmad > depol_scan, Ei=17e9, Ef=18e9, scale_rf=T, file="esr-main.bmad"
...
Bmad > track, init="esr-ltt-bmad.init", qrsh=T, cores=32
```