Design and Applications of the Bmad Library for the Simulation of Particle Beams and X-Rays

David Sagan
Cornell Laboratory for
Accelerator-Based Sciences and Education

Outline

Talk subject:

- Bmad library for particle & X-ray simulations.

Outline:

- Overview & history.
- Useful features.
- Bmad ecosystem of programs.
- ERL & X-ray simulations
- Future plans.

Overview

Overview:

- Written in Fortran. Object oriented from the ground up:

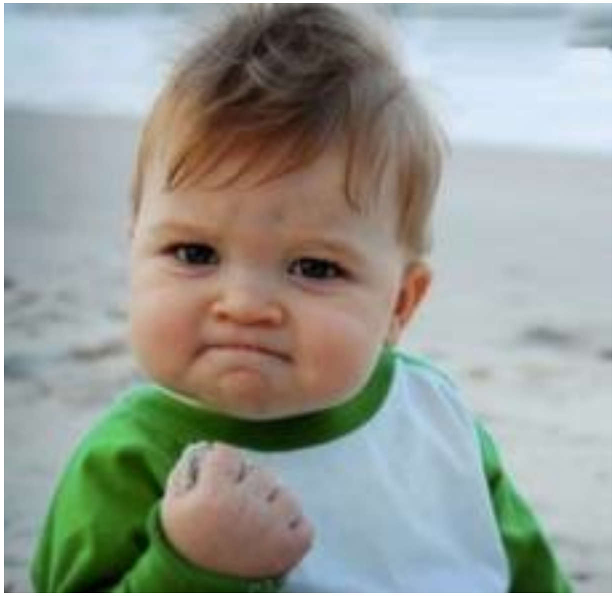
```
type (lat_struct) lat
call bmad_parser ('lat.bmad', lat)
```

- Has structure translation code for interfacing with C++.
- MAD like lattice syntax.
- Open Source http://www.lepp.cornell.edu/~dcs/bmad/

In the Beginning…

Brief History:

- Born at Cornell in mid 1990's
- Started life as modest project: Just wanted to calculate Twiss functions and closed orbit.
- Initially Bmad used a subset of the MAD lattice syntax. Hence the name: "Baby MAD" or "Bmad" for short.



Over the years Bmad had evolved…

And Baby Grows Up...

Currently:

- ~1,000 routines
- ~100,000 lines of code

And it can do much more:

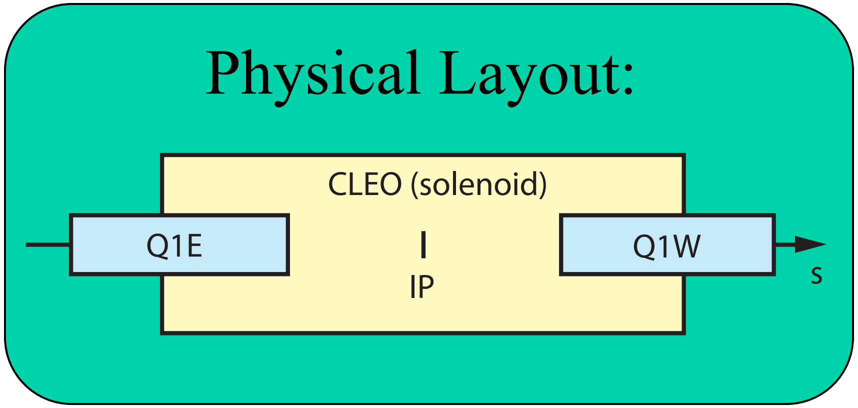
- X-ray simulations
- Coherent synchrotron radiation simulations
- Spin tracking
- HOM studies
- Beam breakup simulations in ERLs
- Intra-beam scattering (IBS) simulations
- Touschek lifetime
- Frequency map analysis
- Dark current tracking
- Etc., etc.

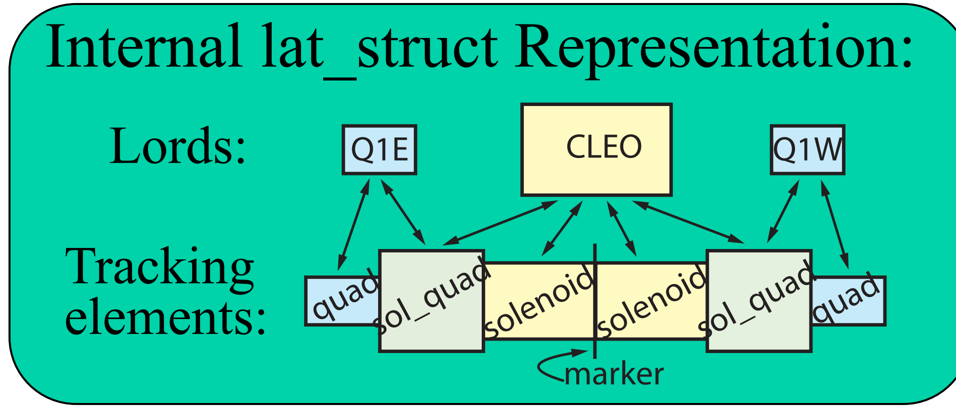
Bmad Features

Bmad has a number of features that over the years have proven useful. Among these are:

- Superposition – Define overlapping elements.
- Controllers – Elements controlling attributes of other elements.
- Element-by-element selection of the tracking method:
    - **bmad_standard** Fast, nonsymplectic
    - **symp_lie_ptc** Symplectic tracking
    - **taylor** Taylor map
    - **linear** Linear tracking
    - **custom** Tracking with custom code
    - **etc.**

Superposition





Superposition allows element overlap. In the lattice file:

cesr: line = (... q1e, dft, ip, dft, q1w ...)
cleo: solenoid, l = 3.5, superimpose, ref = ip

And Bmad does the bookkeeping…

Simplifies life for both user and programmer:

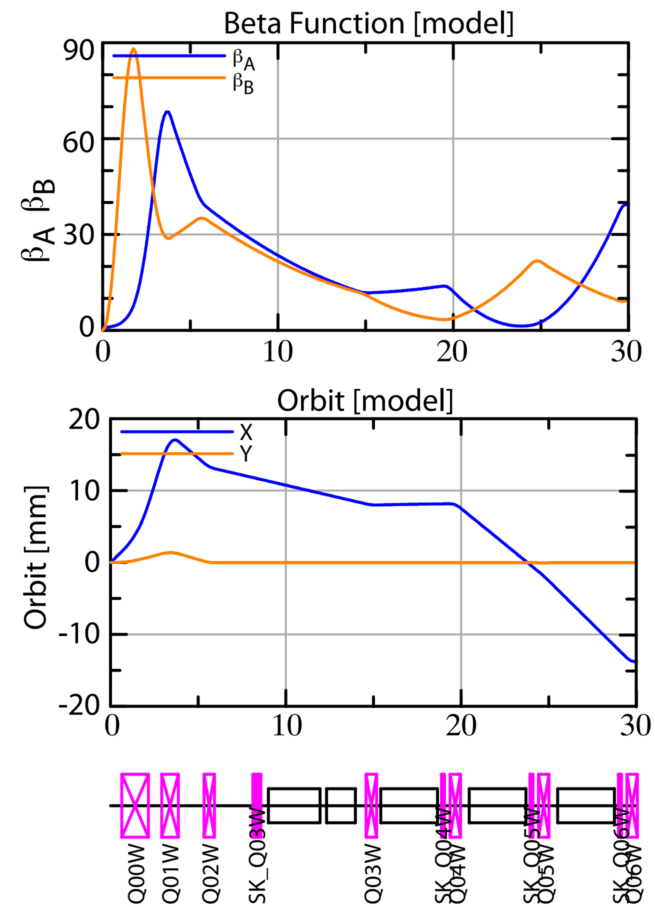
- Simplifies lattice file construction.
- Simplifies varying element attributes in a program.

Bmad Ecosystem

Due to its flexibility, Bmad has been used in a number of programs including:

- **tao** General purpose design and simulation.
- **synrad3d** 3D tracking of synch photons, including reflections, within the beam chamber.
- **cesrv** On-line data taking, simulation, and machine correction for CESR.
- **dark_current_tracker** Dark current electron simulation.
- **freq_map** Frequency map analysis.
- **ibs_sim** Analytic intra-beam scattering (IBS) calculation.
- **touschek_track** Tracking of Touschek particles.

Code reuse: Modules developed for one program can, via Bmad, be used in other programs.



Tao plotting

Dark Current Tracker Program

Problem: Simulate dark current electrons generated at the walls of the beam chamber.

Challenges:

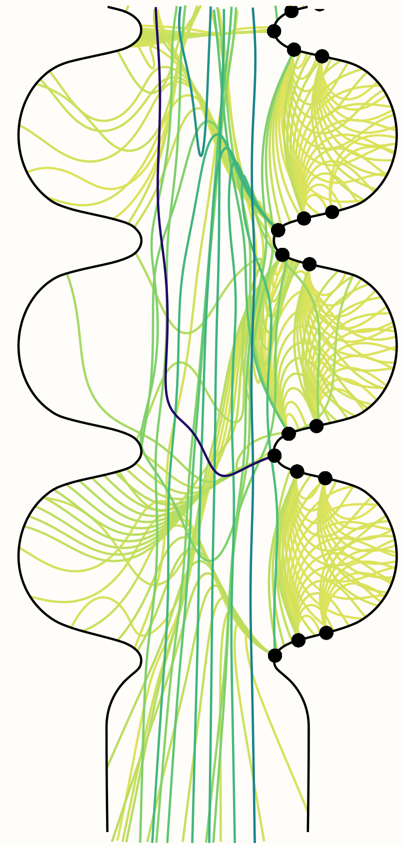
1. Define the beam chamber walls.
2. Be able to track particles that reversed direction longitudinally.

Solutions:

1. X-ray capillary wall code extended for simulating beam chamber walls.
2. Developed time based tracker module.

Result: A useful program was developed and Bmad gets extended capabilities which can then be used in other programs.

Dark current tracks in an RF cavity



Tao: Tool for Accelerator Optics

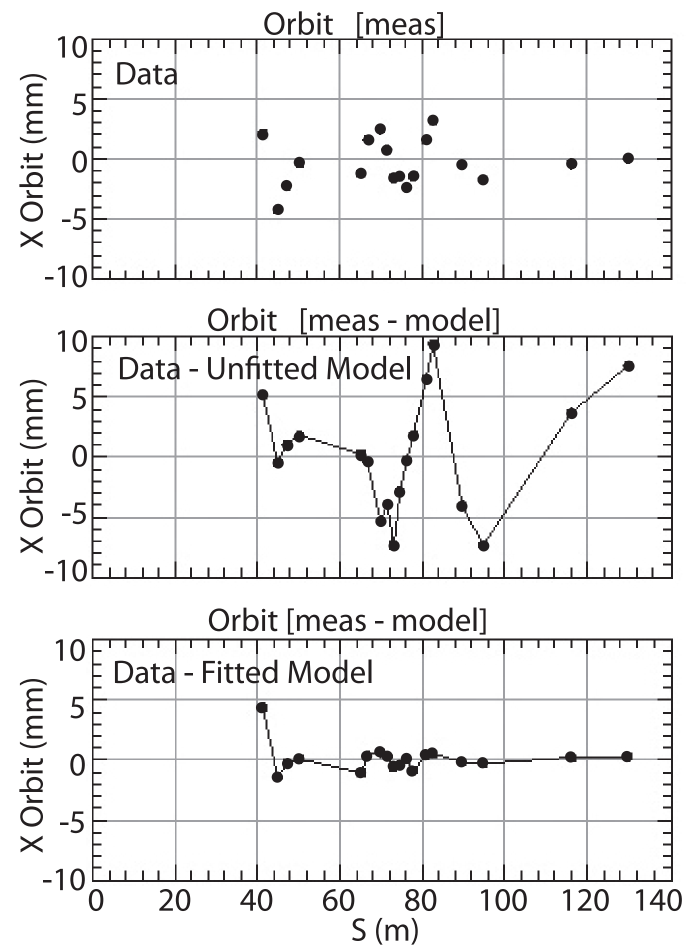
Problem: Bmad is not a program so it cannot be used “out of the box.” for simple calculations.

Solution: Create Tao, a general purpose simulation & design program

- Nonlinear last squares fitting.
- Plotting.
- Twiss and orbit calculations, etc.

Additionally: Tao’s object oriented coding makes it relatively easy to extend it.

- For example: Can add custom commands to interface Tao with a control system.



JLab FEL Modeling

Tao with Bmad gives the flexibility of a library with the convenience of a program.

ERL Simulations

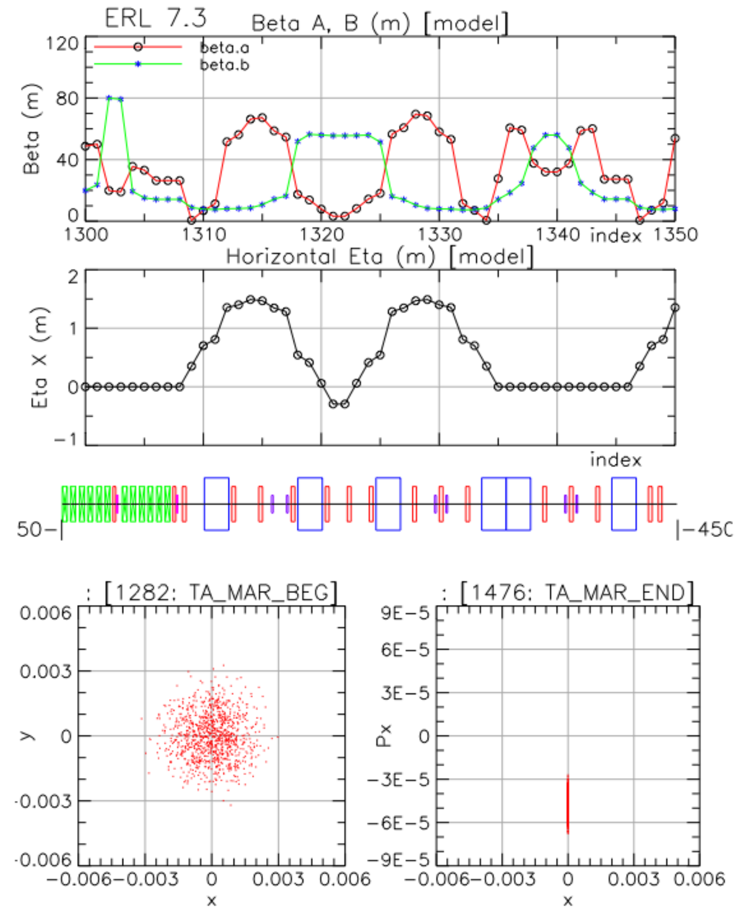
One area of current Bmad development is a unified ERL simulation framework.

Idea: To be able to simulate

- Electrons from the gun cathode to X-ray generation in wigglers and undulators through to the dump.
- X-rays from generation through to the experimental end stations.

Areas of development:

- Full description of the machine.
- Low energy tracking.
- X-ray generation.
- Tracking of X-rays.

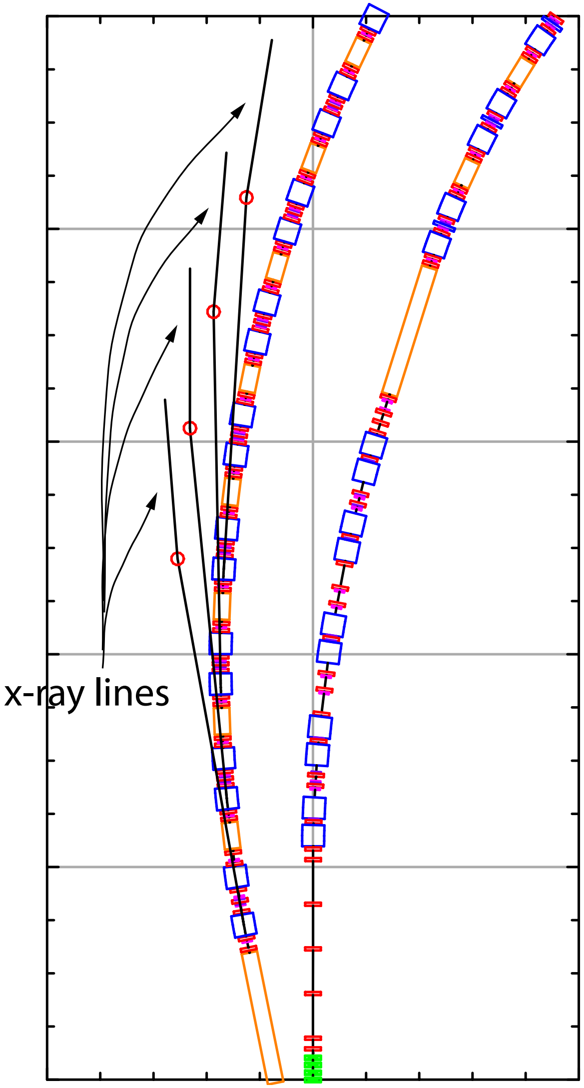


Branching

For defining X-ray and beam dump lines:
Branch and **photon_branch** elements which mark the beginning of a line

```
br1: photon_branch, superimpose, &
      ref = und1, to = tgmono
tgmono: line = (...)  ! Define X-ray line
```

Branch lines can themselves branch.
=> One lattice can hold the "full" machine description.



Low Energy Simulations

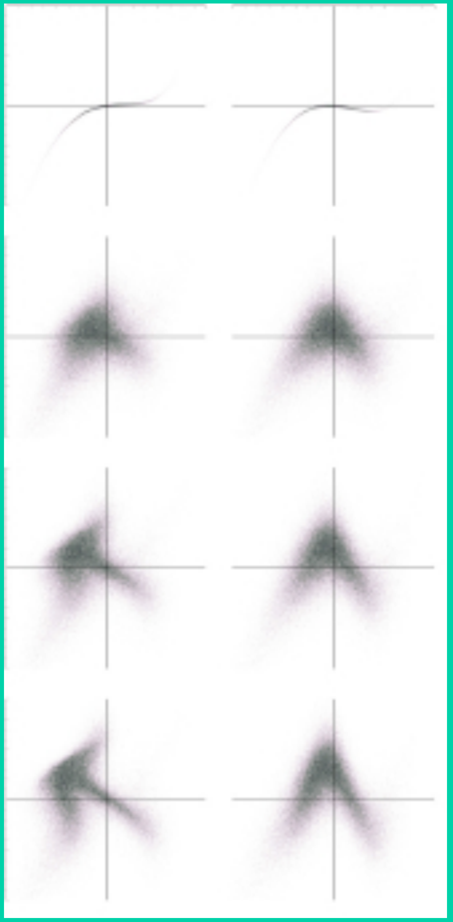
Have developed new lattice elements to handle low energy tracking

- **e_gun** Gun cathode region element.
- **em_field** General field element.

Status:

- Coherent synchrotron radiation model implemented.
- Space Charge: Do not want to reinvent the wheel. Integration with existing SC codes ongoing:
  - Impact-T (Robert Ryne, Ji Qiang)
  - OPAL (Andreas Adelmann)

Low E CSR Simulation
Cornell ERL



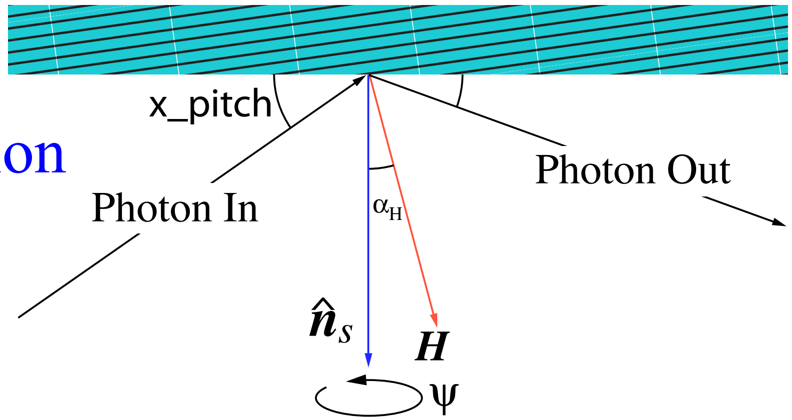
X-Ray Simulation

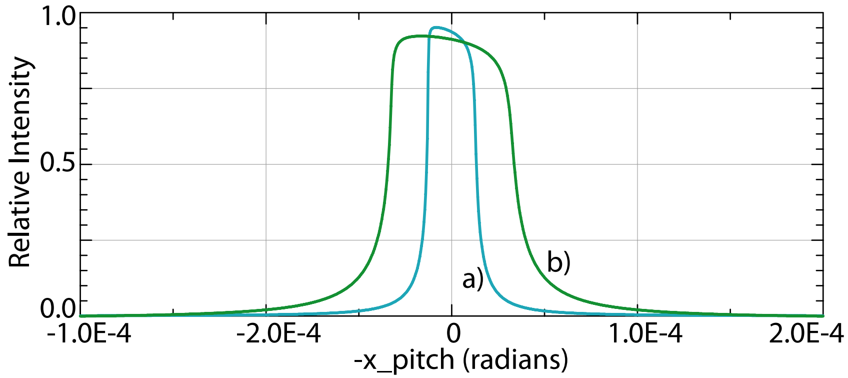
X-ray tracking elements developed:

- Crystal ! Bragg & Laue diffraction
- Capillary ! Focus X-rays
- Mirror
- Multilayer Mirror

Status:

- Bend and Wiggler X-ray generation implemented.
- Tracking needs more debugging.
- First “real world” simulations beginning.





Bragg crystal diffraction

The Future

The evolution of Bmad shows no sign of abating.

Short term:

- Tighter integration with PTC

```
type (lat_struct) lat
call bmad_parser (file_name, lat)
call lat_to_ptc_layout (lat)
```

Long term plans include:

- Integration with the **Shadow** X-ray tracking code.
- Partially coherent X-ray tracking.
- Undulator X-ray generation.
- Nonlinear controllers
- ???

Conclusion

- Bmad has been used successfully at Cornell for a number of years.
- With Bmad, Graduate students can do simulations that would be hard or impossible to do previously.
- Bmad is constantly evolving to meet changing needs.
- Collaborators welcome.
- Caveat: Learning to program with Bmad has a significant learning curve.