Modeling LCLS, LCLS-II: Bmad and Beyond

Christopher Mayes

October 30, 2017

U.S. DEPARTMENT OF ENERGY
Office of Science

SLAC NATIONAL ACCELERATOR LABORATORY

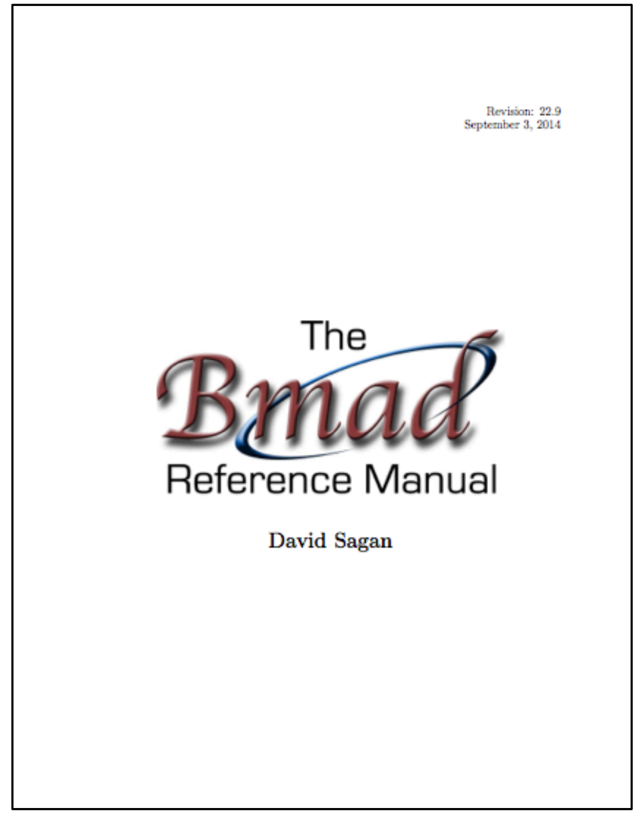
Bmad Overview

SLAC

- Written in Fortran 2008.
- Object-oriented from the ground up.

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

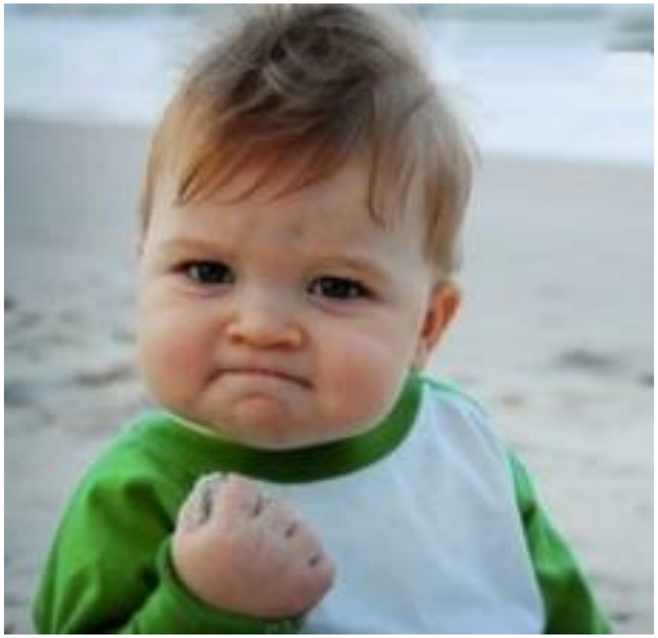
- Has structure translation code for interfacing with C++.
- With certain restrictions, Bmad can be run multi-threaded.
- Lattice files use a MAD like syntax.
- Well documented (Manual is ~500 pages).
- Open Source:
  http://www.lepp.cornell.edu/~dcs/bmad/



In the Beginning…

Brief History of Bmad:

- Bmad is a software toolkit for the simulation of charged particles and X-rays.
- Born at Cornell in mid 1990s
- Started life as modest project: Just wanted to calculate Twiss functions and closed orbits.
- 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:

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

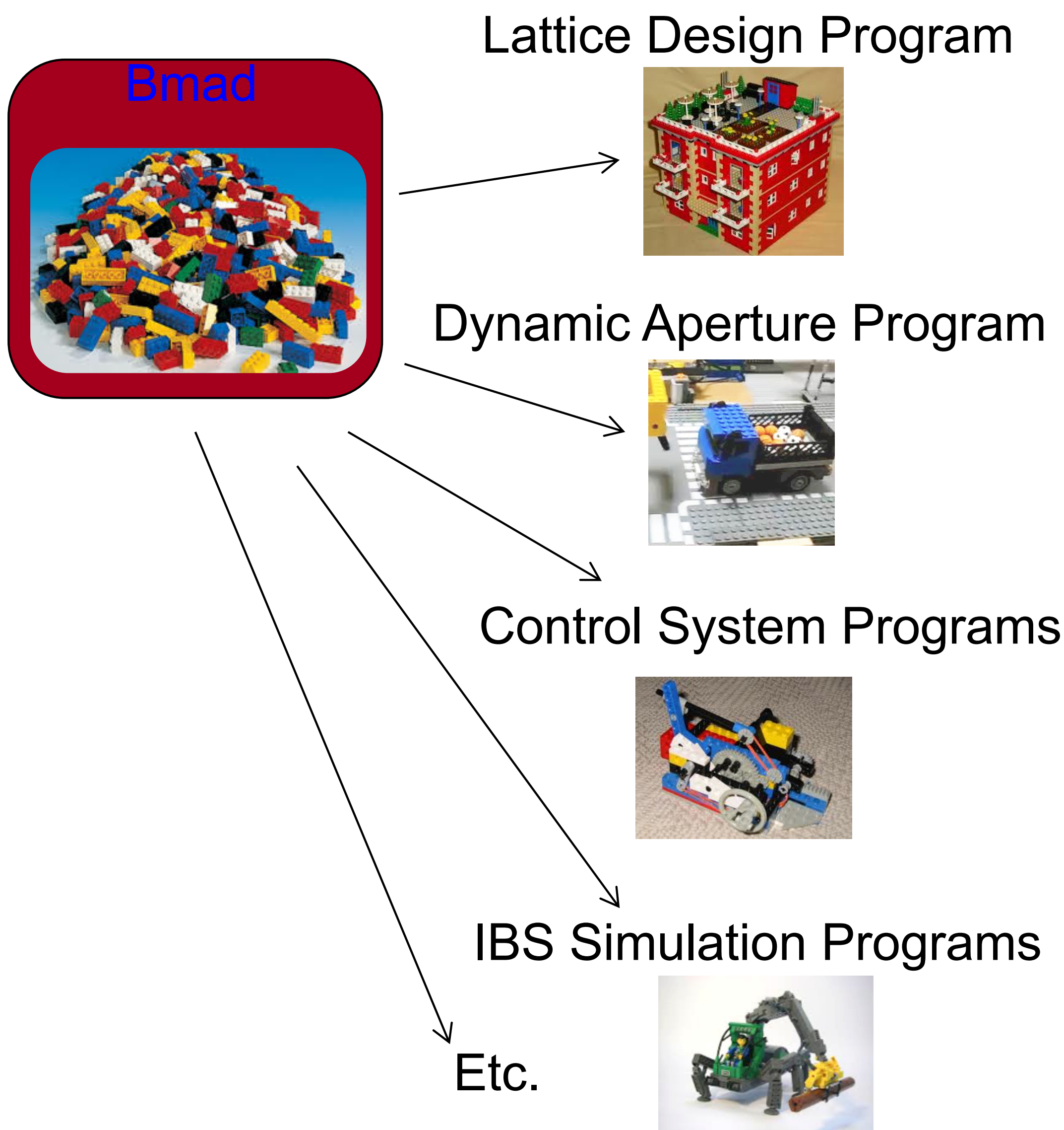
And it can do much more:

- Lattice design
- X-ray simulations
- Spin tracking
- Wakefields and HOMs
- Beam breakup simulations in ERLs
- Intra-beam scattering (IBS) simulations
- Coherent Synchrotron Radiation (CSR)
- Touschek Simulations
- Frequency map analysis
- Dark current tracking
- Etc., etc.

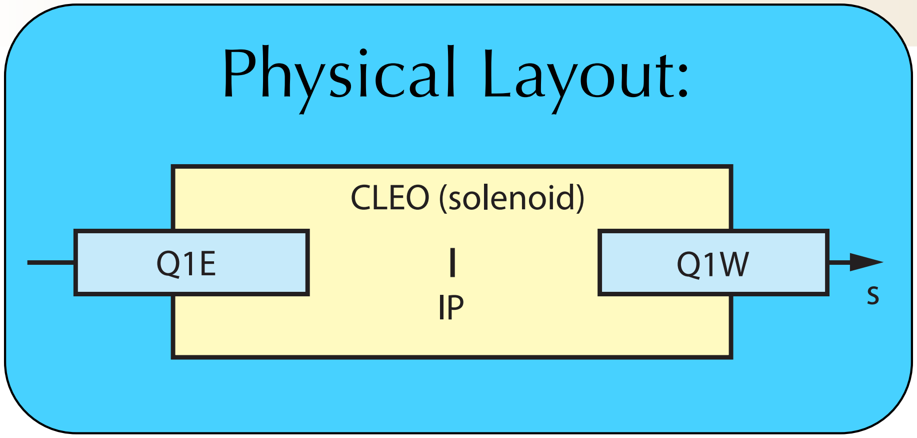
Bmad Philosophy

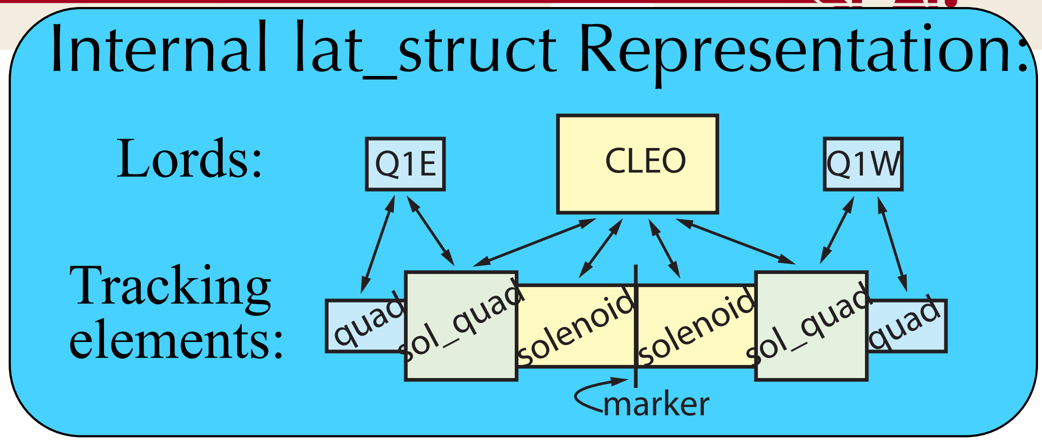
Advantages of a toolkit:

- Cuts down on the time needed to develop programs.
- Cuts down on programming errors (via code reuse).
- Provides a simple mechanism for lattice function calculations from within control system programs.
- Standardizes sharing of lattice information between programs.



Superposition





Superposition allows element overlap. In the lattice file:

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

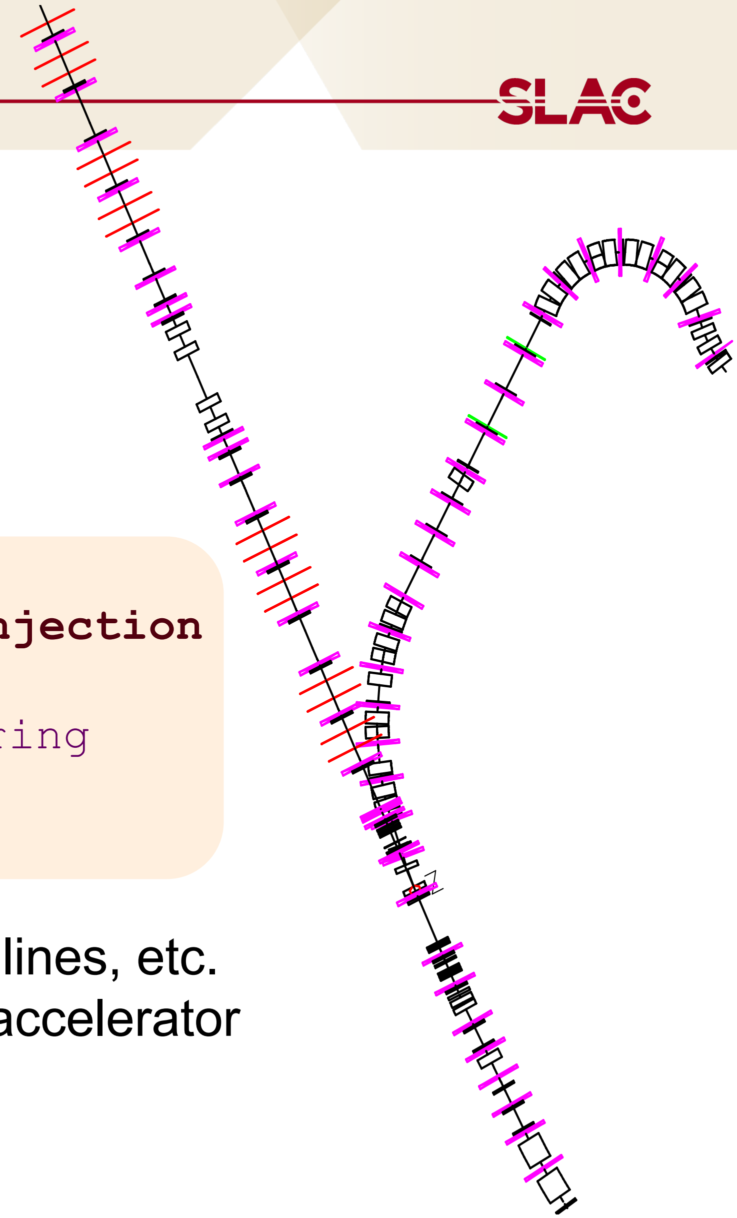
And Bmad does the bookkeeping…

Simplifies life for both user and programmer:

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

Forking

- Bmad can join different beam lines using fork elements.
- Example: SuperKEKB Low Energy Ring (LER) injection

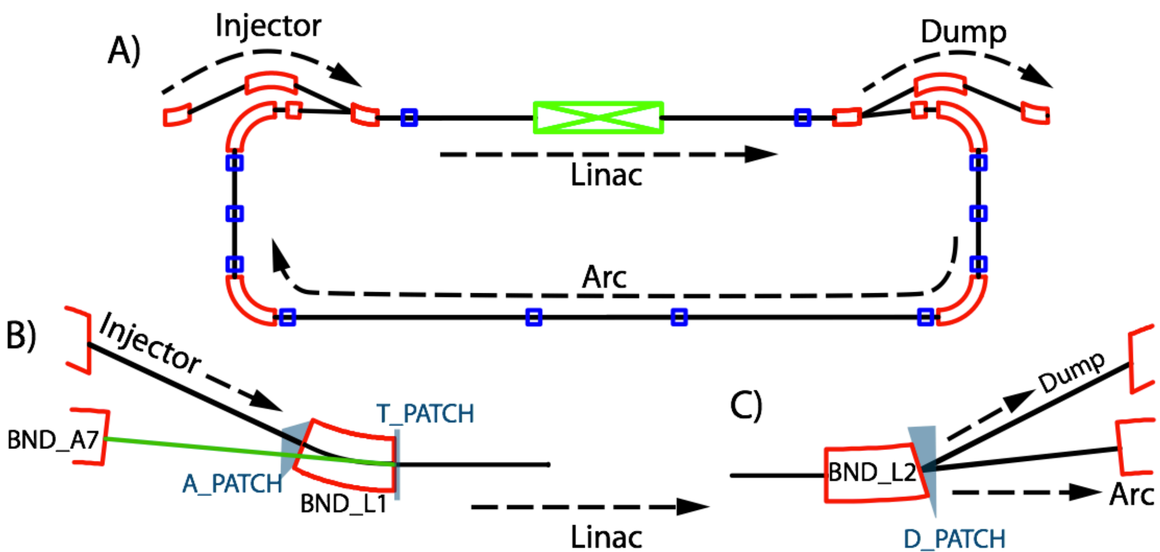


```
to_LER: fork, to_line = ler, to_element = injection
inj: line = (..., to_LER)   ! injection line
LER: line = (..., injection, ...)    ! LER ring
use, inj
```

- Can splice together rings, dump lines, Linacs, X-ray lines, etc.
  ➔ One lattice can hold the description of the entire accelerator complex.

Multipass

Lines having common elements. Example ERL loop:




```
BEND_L1: sbend, angle = -25*pi/180, l = 0.2, ...
BEND_L2: BEND_L1
A_PATCH: patch, flexible = T
D_PATCH: patch, x_offset = -0.034, x_offset = asin(-0.32)
INJECT: line = (...)
LINAC: line[multipass] = (BEND_L1, ..., BEND_L2)
ARC: line = (..., BEND_A7)
DUMP: line = (...)
ERL: line = (INJECT, LINAC, ARC, A_PATCH, LINAC, D_PATCH, DUMP)
```

Tracking method selection

Can set how each element is tracked:

- `bmad_standard` Fast, nonsymplectic
- `symp_lie_ptc` Symplectic tracking using PTC
- `taylor` Taylor map tracking using PTC
- `linear` Linear tracking
- `runge_kutta` Track through fields.
- `custom` Tracking with custom code
- …

```
my_ele: em_field, tracking_method = runge_kutta, field_calc=custom, ...
```

Advantages:

- Enables simulation of unique element types
- Can compare different tracking methods

All elements and tracking methods

Element Class	Bmad_Standard	Boris	Custom	Linear	MAD	Runge_Kutta	Symp_Lie_Bmad	Symp_Lie_PTC	Symp_Map	Taylor	Time_Runge_Kutta
ab_multipole	D		X	X				X	X	X	
beambeam	D		X	X							
bend_sol_quad			X				D				
capillary	D		X								
crystal	D		X								
custom		X	D	X		X					X
drift	D	X	X	X	X	X		X	X	X	X
e_gun			X								D
ecollimator	D	X	X	X		X		X	X	X	X
elseparator	D	X	X	X	X	X		X	X	X	X
em_field		X	X			D					X
floor_shift	D		X								
hkicker	D	X	X	X		X		X	X	X	X
instrument	D	X	X	X		X		X	X	X	X
kicker	D	X	X	X		X		X	X	X	X
lcavity	D	X	X	X		X		X	X	X	X
marker	D		X	X				X	X	X	
match	D		X							*	
monitor	D	X	X	X		X		X	X	X	X
mirror	D		X								
multipole	D		X	X				X	X	X	
multilayer	D		X								
octupole	D	X	X	X		X		X	X	X	X
patch	D		X			X		X		X	
quadrupole	D	X	X	X	X	X	X	X	X	X	X
rbend	D		X	X	X			X	X	X	
rcollimator	D	X	X	X		X		X	X	X	X
rfcavity	D	X	X	X	X	X		X	X	X	X
sad_mult	D		X	X				X	X	X	
sample	D		X								
sbend	D		X	X	X	X		X	X	X	X
sextupole	D	X	X	X	X	X		X	X	X	X
solenoid	D	X	X	X	X	X	X	X	X	X	X
sol_quad	D	X	X	X		X	X	X	X	X	X
taylor	X		X	X						D	
vkicker	D	X	X	X		X		X	X	X	X
wiggler (map type)		X	X	X		X	X	X	X	X	X
wiggler (periodic type)	D	X	X	X		X[a]	X[a]	X[a]	X[a]	X[a]	

[a]See §3.43.2 for more details

Table 5.1: Table of available **tracking_method** switches for a given element class. "D" denotes the default method. "X" denotes an available method. "*" denotes that the Taylor map will only be first order.

Spin	Bmad_Standard	Custom	Symp_Lie_PTC	Tracking
ab_multipole	D	X		
beambeam		X		
bend_sol_quad	D	X	X	X
capillary				
crystal				
custom		D		X
drift	D	X	X	X
e_gun		X		D
ecollimator	D	X	X	X
elseparator	D	X	X	X
em_field		X		D
hkicker	D	X	X	X
instrument	D	X	X	X
kicker	D	X	X	X
lcavity	D	X	X	X
marker	D	X	X	X
match		X		
mirror				
monitor	D	X	X	X
multipole	D	X	X	
multilayer				
octupole	D	X	X	X
patch		X	X	
quadrupole	D	X	X	X
rbend	D	X	X	X
rcollimator	D	X	X	X
rfcavity	D	X	X	X
sad_mult		X		
sample				
sbend	D	X	X	X
sextupole	D	X	X	X
solenoid	D	X	X	X
sol_quad	D	X	X	X
taylor				
vkicker	D	X	X	X
wiggler	D	X	X	X

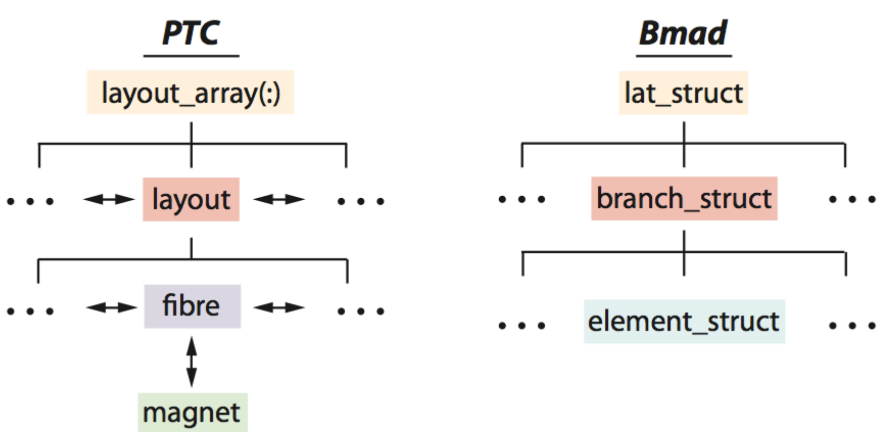
Table 5.3: Table of available **spin_tracking_method** switches for a given element class. "D" denotes the default method. "X" denotes an available method.

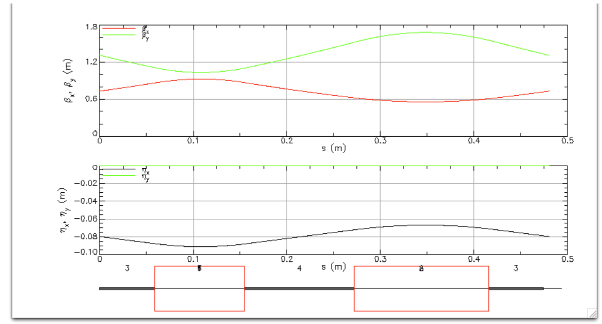
Etienne Forest's FPP/PTC

Etienne Forest's FPP/PTC simulation toolkit:

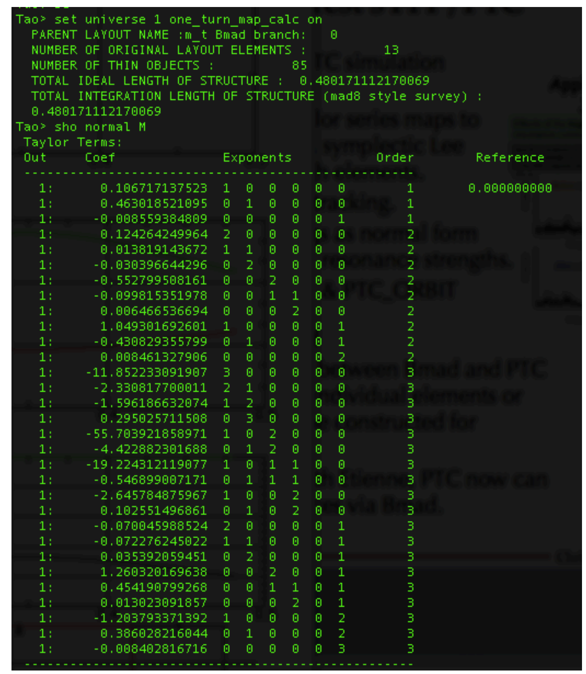
- Taylor maps to arbitrary order via symplectic integration through elements.
- Spin tracking.
- Normal form analysis.
- Amplitude Dependent Spin Tune and Invariant Spin Field calculations.
- Also used in MAD-X & PTC_ORBIT

Interface routines between Bmad and PTC allow tracking of individual elements or PTC lattices can be constructed for analysis.





FFAG cell



```
Tao> set universe 1 one_turn_map_calc on
  PARENT LAYOUT NAME :m_t Bmad branch:   0
  NUMBER OF ORIGINAL LAYOUT ELEMENTS :          13
  NUMBER OF THIN OBJECTS :            85
  TOTAL IDEAL LENGTH OF STRUCTURE :  0.480171112170069
  TOTAL INTEGRATION LENGTH OF STRUCTURE (mad8 style survey) :
  0.480171112170069
Tao> sho normal M
 Taylor Terms:
 Out     Coef                Exponents           Order        Reference
 ---------------------------------------------------------
   1:      0.106717137523  1  0  0  0  0  0          1       0.000000000
   1:      0.463018521095  0  1  0  0  0  0          1
   1:     -0.008559384809  0  0  0  0  0  1          1
   1:      0.124264249964  2  0  0  0  0  0          2
   1:      0.013819143672  1  1  0  0  0  0          2
   1:     -0.030396644296  0  2  0  0  0  0          2
   1:     -0.552799508161  0  0  2  0  0  0          2
   1:     -0.099815351978  0  0  1  1  0  0          2
   1:      0.006466536694  0  0  0  2  0  0          2
   1:      1.049301692601  1  0  0  0  0  1          2
   1:     -0.430829355799  0  1  0  0  0  1          2
   1:      0.008461327906  0  0  0  0  0  2          2
   1:    -11.852233091907  3  0  0  0  0  0          3
   1:     -2.330817700011  2  1  0  0  0  0          3
   1:     -1.596186632074  1  2  0  0  0  0          3
   1:      0.295025711508  0  3  0  0  0  0          3
   1:    -55.703921858971  1  0  2  0  0  0          3
   1:     -4.422882301688  0  1  2  0  0  0          3
   1:    -19.224312119077  1  0  1  1  0  0          3
   1:     -0.546899007171  0  1  1  1  0  0          3
   1:     -2.645784875967  1  0  0  2  0  0          3
   1:      0.102551496861  0  1  0  2  0  0          3
   1:     -0.070045988524  2  0  0  0  0  1          3
   1:     -0.072276245022  1  1  0  0  0  1          3
   1:      0.035392059451  0  2  0  0  0  1          3
   1:      1.260320169638  0  0  2  0  0  1          3
   1:      0.454190799268  0  0  1  1  0  1          3
   1:      0.013023091857  0  0  0  2  0  1          3
   1:     -1.203793371392  1  0  0  0  0  2          3
   1:      0.386028216044  0  1  0  0  0  2          3
   1:     -0.008402816716  0  0  0  0  0  3          3
 ---------------------------------------------------------
```

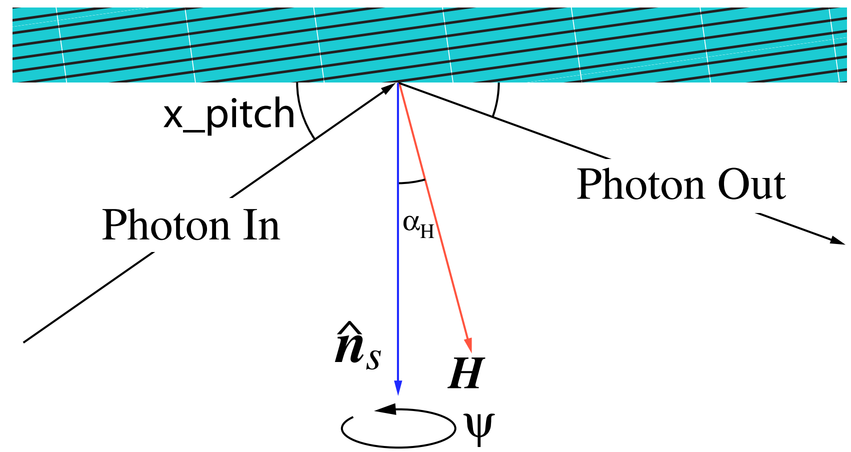
One turn map calc in Tao

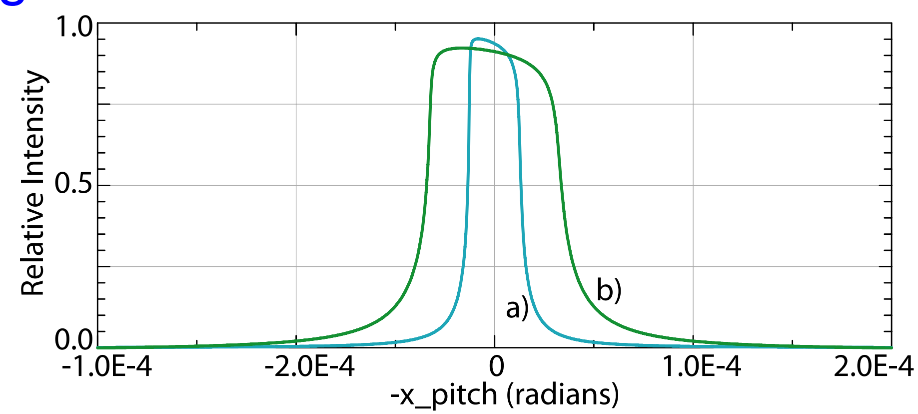
X-ray simulation

Both incoherent and coherent ray-tracing implemented (but needs more testing).

X-ray tracking elements developed:

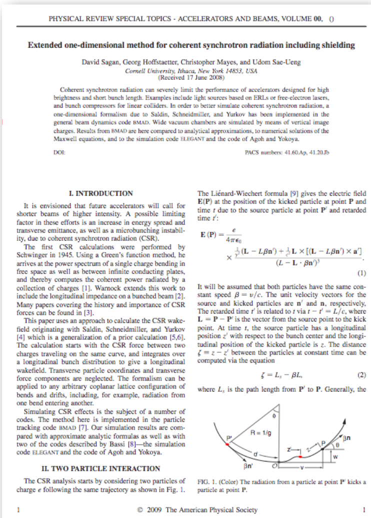
- Crystal ! Bragg & Laue diffraction
- Capillary ! Focus X-rays
- Mirror
- Multilayer Mirror
- Diffraction_plate ! Apertures, Zone plates





Bragg crystal diffraction simulation

CSR with Shielding


PHYSICAL REVIEW SPECIAL TOPICS - ACCELERATORS AND BEAMS, VOLUME 00, 0

Extended one-dimensional method for coherent synchrotron radiation including shielding

David Sagan, Georg Hoffstaetter, Christopher Mayes, and Udom Sae-Ueng
Cornell University, Ithaca, New York 14853, USA
(Received 17 June 2008)

Coherent synchrotron radiation can severely limit the performance of accelerators designed for high brightness and short bunch length. Examples include light sources based on ERLs or free-electron lasers, and bunch compressors for linear colliders. In order to better simulate coherent synchrotron radiation, a one-dimensional formalism due to Saldin, Schneidmiller, and Yurkov has been implemented in the general beam dynamics code BMAD. Wide vacuum chambers are simulated by means of vertical image charges. Results from BMAD are here compared to analytical approximations, to numerical solutions of the Maxwell equations, and to the simulation code ELEGANT and the code of Agoh and Yokoya.

I. INTRODUCTION

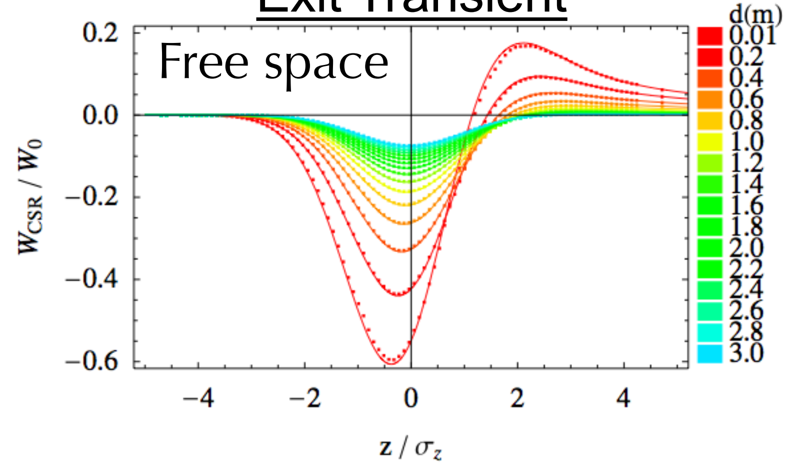
II. TWO PARTICLE INTERACTION

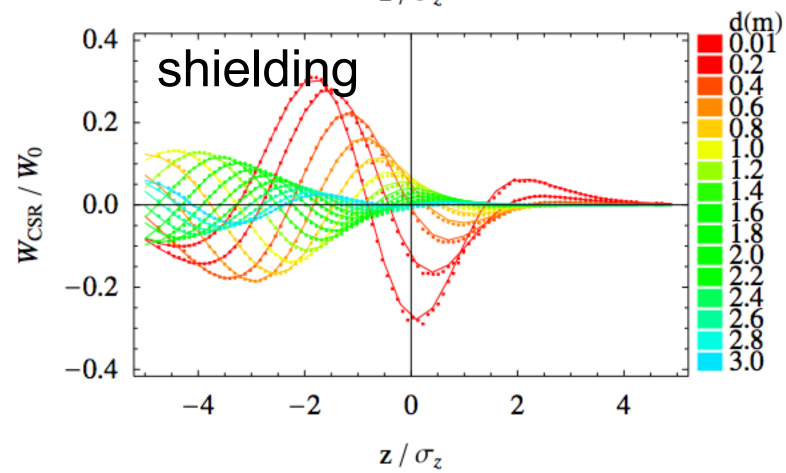
© 2009 The American Physical Society

We developed and benchmarked a 1-D CSR simulation technique with shielding by the vacuum chamber in Bmad

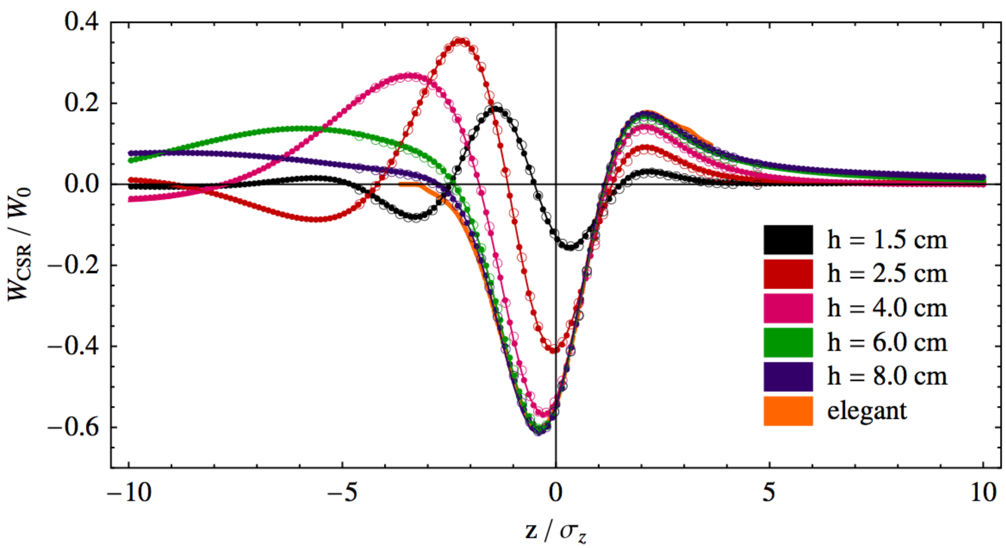
[Sagan, Hoffstaetter, Mayes, Sae-Ueng PRST-AB **12**, 040703 (2009)]

Exit Transient





Compared with Agoh & Yokoya



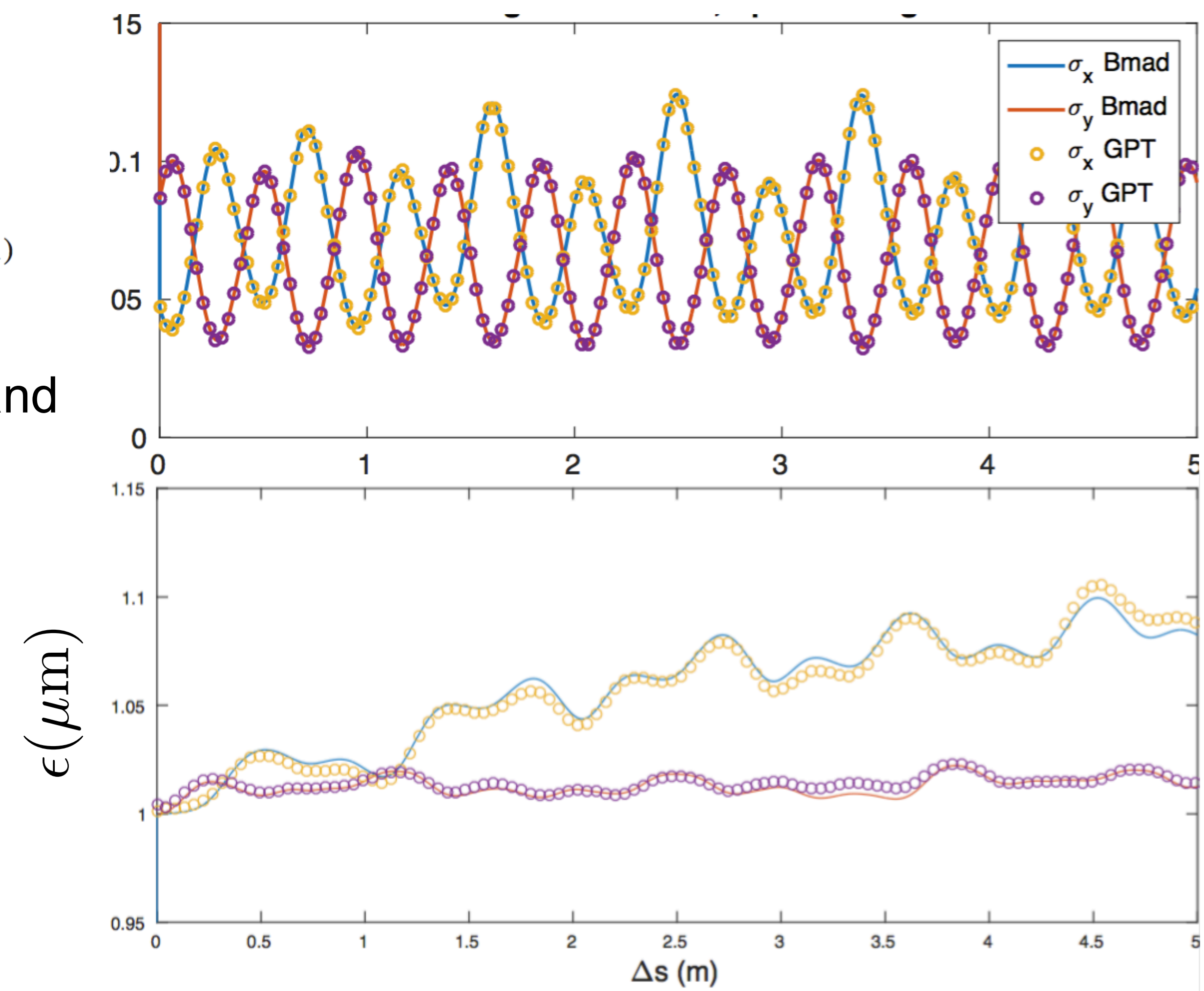
Transverse Space Charge

Also in the 2009 paper was an excellent approximation of the space charge kick due to a Gaussian slice

$$dK_{\mathrm{SC}} \approx \frac{r_c mc^2 \mathrm{sign}(\zeta)\rho(z')dz'}{\sigma_x\sigma_y \exp[\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2}] + \frac{\sigma_x^2+\sigma_y^2}{\sigma_x+\sigma_y}\gamma|\zeta| + \gamma^2\zeta^2}. \quad (31)$$

This includes both longitudinal and transverse space charge

1 nC bunch through straight FFAG section at 42 MeV

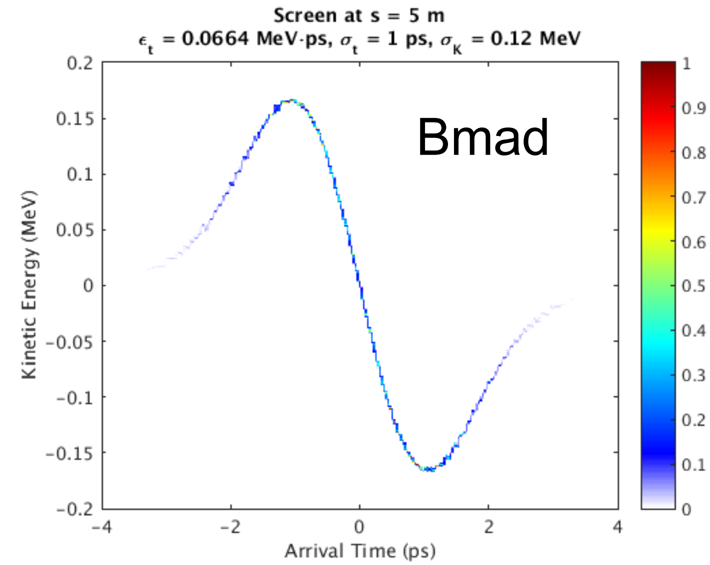


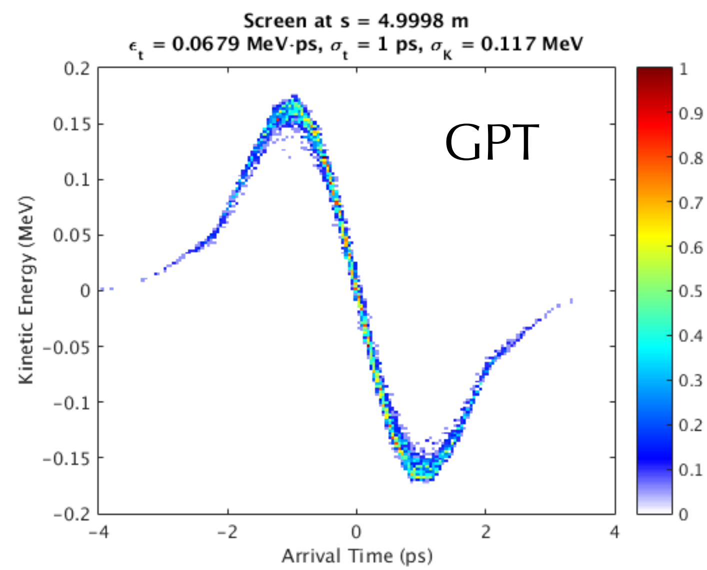
Longitudinal Space Charge

Also in the 2009 paper was an excellent approximation of the space charge kick due to a Gaussian slice

$$dK_{\mathrm{SC}} \approx \frac{r_c mc^2 \mathrm{sign}(\zeta)\rho(z')dz'}{\sigma_x\sigma_y \exp[\frac{x^2}{2\sigma_x^2} + \frac{y^2}{2\sigma_y^2}] + \frac{\sigma_x^2+\sigma_y^2}{\sigma_x+\sigma_y}\gamma|\zeta| + \gamma^2\zeta^2}. \quad (31)$$

This includes both longitudinal and transverse space charge



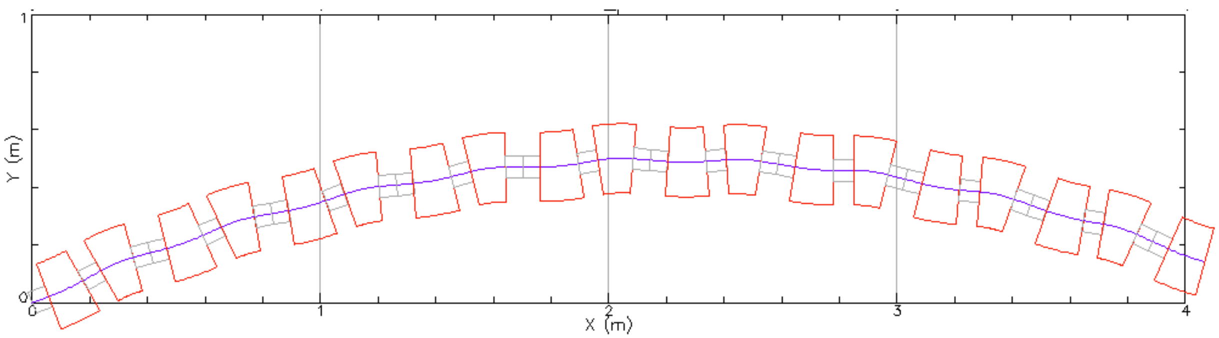


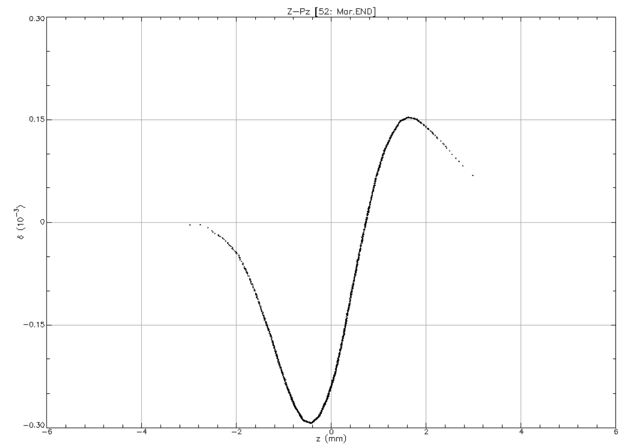
Page Headline

More general 1-D CSR model

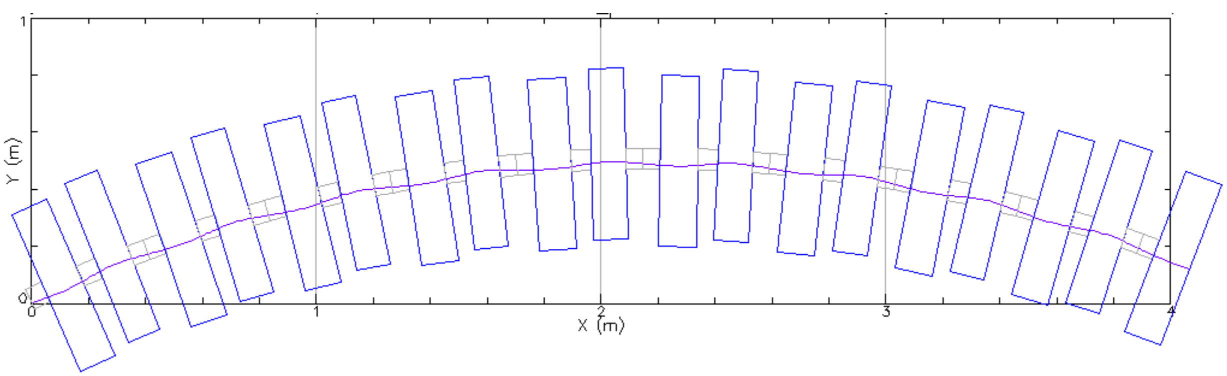
- The old code and 2009 paper assumed that the beam followed the reference coordinate system
- The new code has been generalized to deal with arbitrary orbit.
- IPAC17:THPAB076

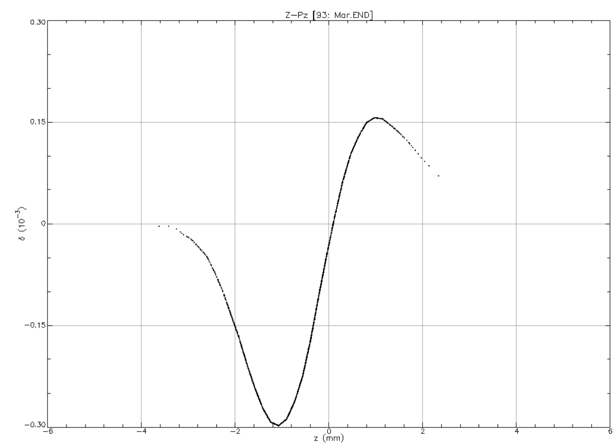
Beam follows reference (bends with quad moment)





Beam orbit with offset quads





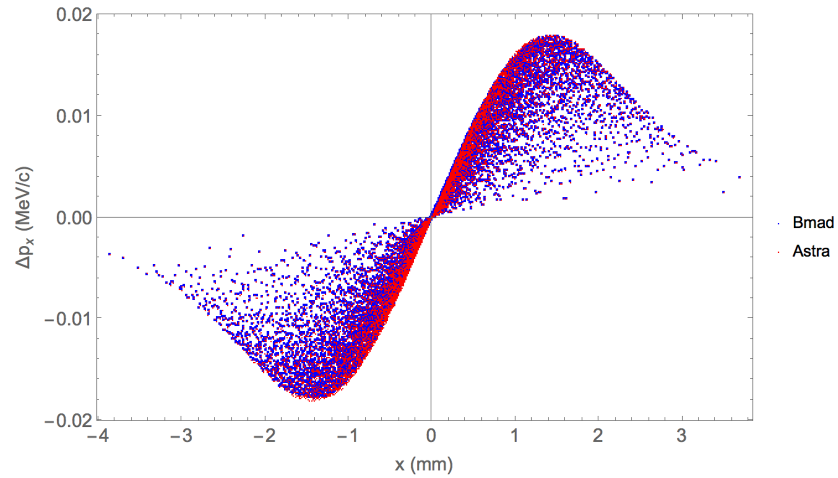
Page Headline

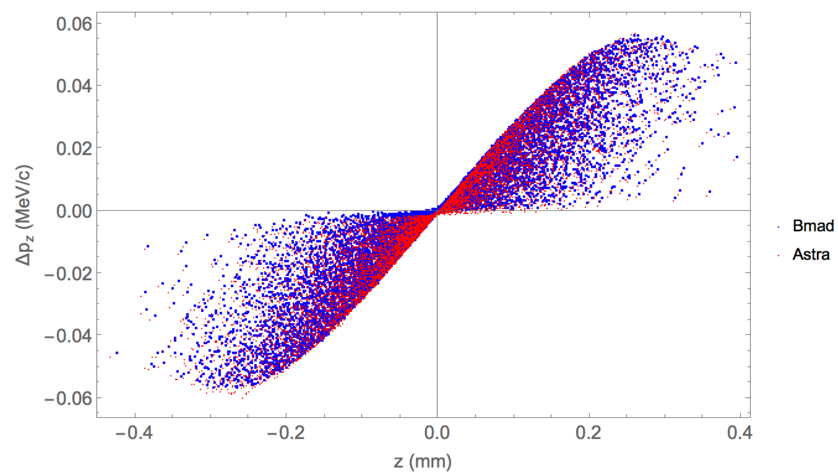
NEW: 3D spacecharge

SLAC

- Robert Ryne (LBNL, original author of IMPACT) has developed a new stand-alone space charge package.
- Parallelized via MPI
- Open source

- New collaboration to generalize its use for easy integration into other codes.
- Currently incorporating it into Bmad
- Adding OpenMP parallelization.
- Benchmarks well with Astra, ImpactZ, MaryLie/IMPACT

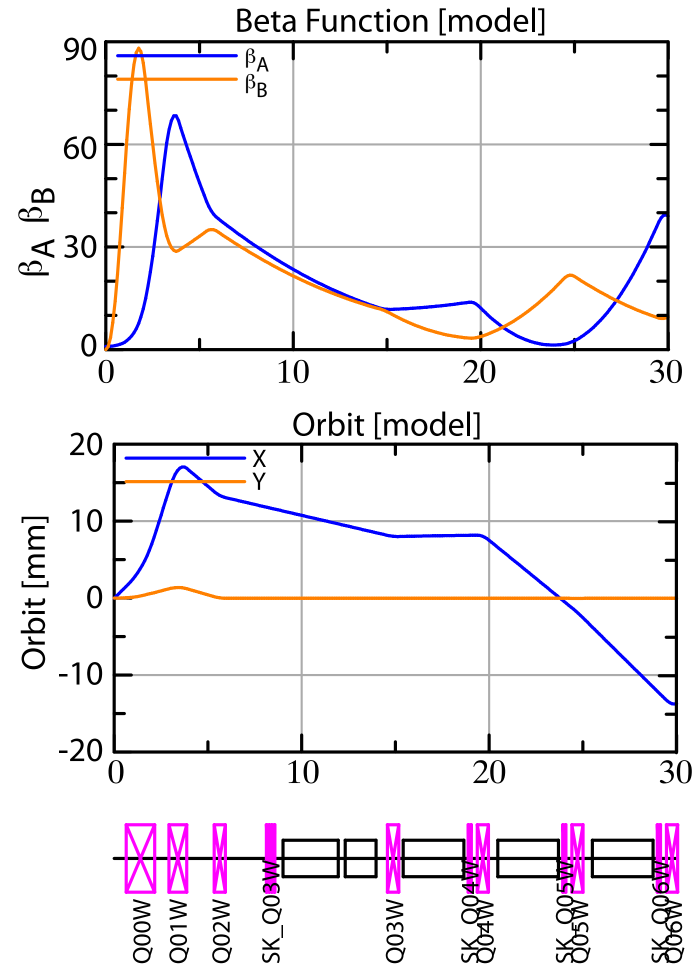




Bmad Ecosystem

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

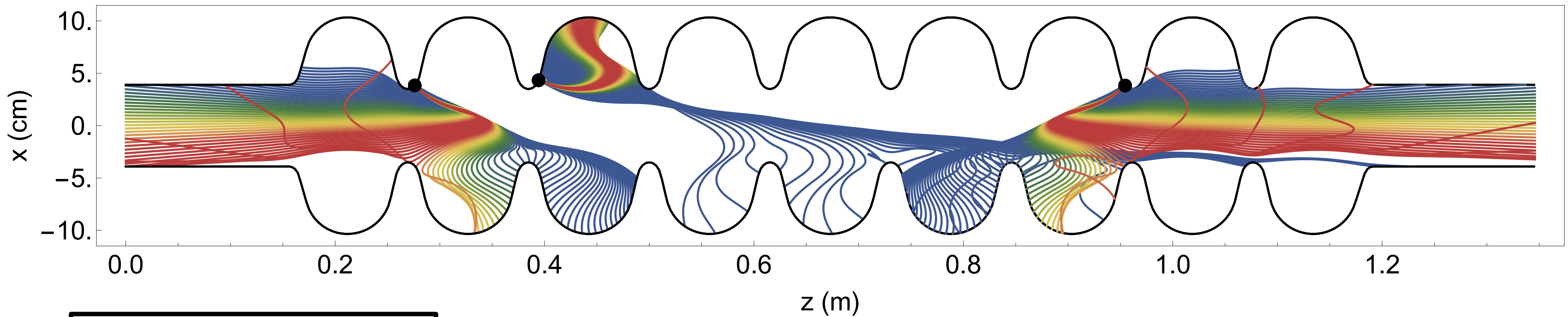
- **tao** General purpose design and simulation.
- **synrad3d** 3D tracking of synch photons, including reflections, within the beam chamber.
- **cesrv** On-line data taking, simulation, and machine correction for CESR.
- **dark_current_tracker** Dark current electron simulation.
- **freq_map** Frequency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

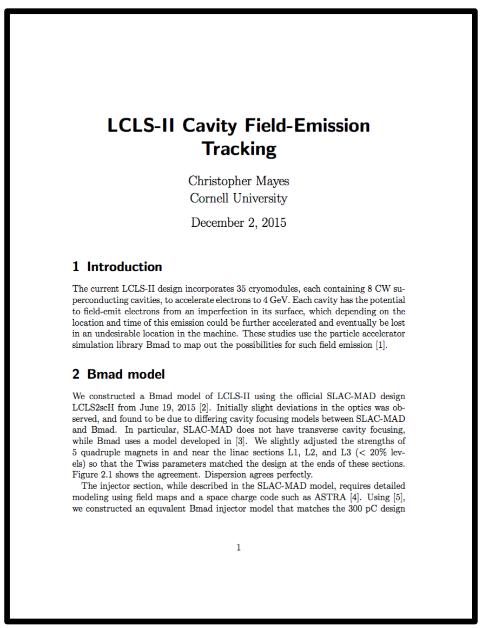


Code reuse: Modules developed for one program can, via Bmad, be used in other programs.

Dark Current Tracking for LCLS2

Field emitters





LCLS-II Cavity Field-Emission Tracking

Christopher Mayes
Cornell University

December 2, 2015

1 Introduction

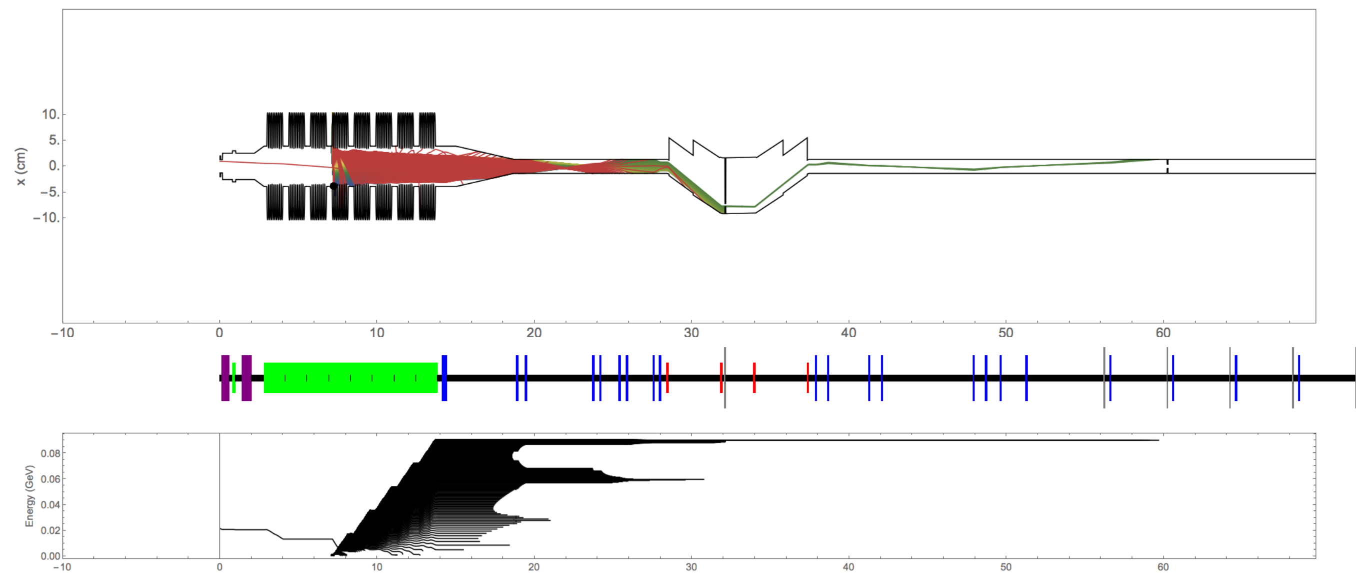
The current LCLS-II design incorporates 35 cryomodules, each containing 8 CW superconducting cavities, to accelerate electrons to 4 GeV. Each cavity has the potential to field-emit electrons from an imperfection in its surface, which depending on the location and time of this emission could be further accelerated and eventually be lost in an undesirable location in the machine. These studies use the particle accelerator simulation library Bmad to map out the possibilities for such field emission [1].

2 Bmad model

We constructed a Bmad model of LCLS-II using the official SLAC-MAD design LCLS2scH from June 19, 2015 [2]. Initially slight deviations in the optics was observed, and found to be due to differing cavity focusing models between SLAC-MAD and Bmad. In particular, SLAC-MAD does not have transverse cavity focusing, while Bmad uses a model developed in [3]. We slightly adjusted the strengths of 5 quadruple magnets in and near the linac sections L1, L2, and L3 (< 20% levels) so that the Twiss parameters matched the design at the ends of these sections. Figure 2.1 shows the agreement. Dispersion agrees perfectly.

The injector section, while described in the SLAC-MAD model, requires detailed modeling using field maps and a space charge code such as ASTRA [4]. Using [5], we constructed an equvalent Bmad injector model that matches the 300 pC design

1



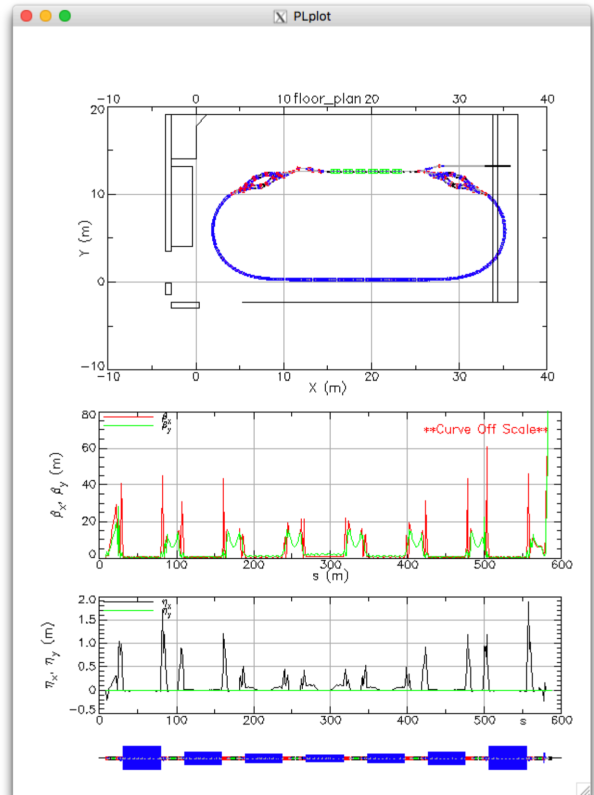
https://www.classe.cornell.edu/~cem52/LCLS2/2015_09_16-LCLS2_dark_current.pdf

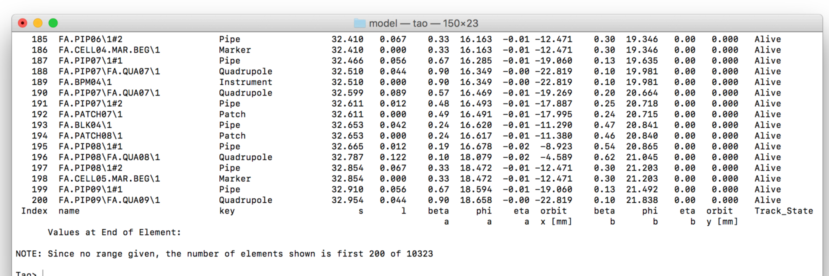
Tao: Tool for Accelerator Optics

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

Solution: Develop Tao - a general purpose simulation & design program with
- Twiss and orbit calculations.
- Nonlinear optimization.
- Analysis of complicated geometries.
- Etc.

Additionally: Tao’s object oriented coding makes it relatively easy to extend it.
- Can add custom commands to interface Tao with a control system.
- Can be driven by Python




model — tao — 150×23

Index	name	key	s	l	beta a	phi a	eta a	orbit x [mm]	beta b	phi b	eta b	orbit y [mm]	Track_State
185	FA.PIP06\1#2	Pipe	32.410	0.067	0.33	16.163	-0.01	-12.471	0.30	19.346	0.00	0.000	Alive
186	FA.CELL04.MAR.BEG\1	Marker	32.410	0.000	0.33	16.163	-0.01	-12.471	0.30	19.346	0.00	0.000	Alive
187	FA.PIP07\1#1	Pipe	32.466	0.056	0.67	16.285	-0.01	-19.060	0.13	19.635	0.00	0.000	Alive
188	FA.PIP07\FA.QUA07\1	Quadrupole	32.510	0.044	0.90	16.349	-0.00	-22.819	0.10	19.981	0.00	0.000	Alive
189	FA.BPM04\1	Instrument	32.510	0.000	0.90	16.349	-0.00	-22.819	0.10	19.981	0.00	0.000	Alive
190	FA.PIP07\FA.QUA07\1	Quadrupole	32.599	0.089	0.57	16.469	-0.01	-19.269	0.20	20.664	0.00	0.000	Alive
191	FA.PIP07\1#2	Pipe	32.611	0.012	0.48	16.493	-0.01	-17.887	0.25	20.718	0.00	0.000	Alive
192	FA.PATCH07\1	Patch	32.611	0.000	0.49	16.491	-0.01	-17.995	0.24	20.715	0.00	0.000	Alive
193	FA.BLK04\1	Pipe	32.653	0.042	0.24	16.620	-0.01	-11.290	0.47	20.841	0.00	0.000	Alive
194	FA.PATCH08\1	Patch	32.653	0.000	0.24	16.617	-0.01	-11.380	0.46	20.840	0.00	0.000	Alive
195	FA.PIP08\1#1	Pipe	32.665	0.012	0.19	16.678	-0.02	-8.923	0.54	20.865	0.00	0.000	Alive
196	FA.PIP08\FA.QUA08\1	Quadrupole	32.787	0.122	0.10	18.079	-0.02	-4.589	0.62	21.045	0.00	0.000	Alive
197	FA.PIP08\1#2	Pipe	32.854	0.067	0.33	18.472	-0.01	-12.471	0.30	21.203	0.00	0.000	Alive
198	FA.CELL05.MAR.BEG\1	Marker	32.854	0.000	0.33	18.472	-0.01	-12.471	0.30	21.203	0.00	0.000	Alive
199	FA.PIP09\1#1	Pipe	32.910	0.056	0.67	18.594	-0.01	-19.060	0.13	21.492	0.00	0.000	Alive
200	FA.PIP09\FA.QUA09\1	Quadrupole	32.954	0.044	0.90	18.658	-0.00	-22.819	0.10	21.838	0.00	0.000	Alive

Values at End of Element:

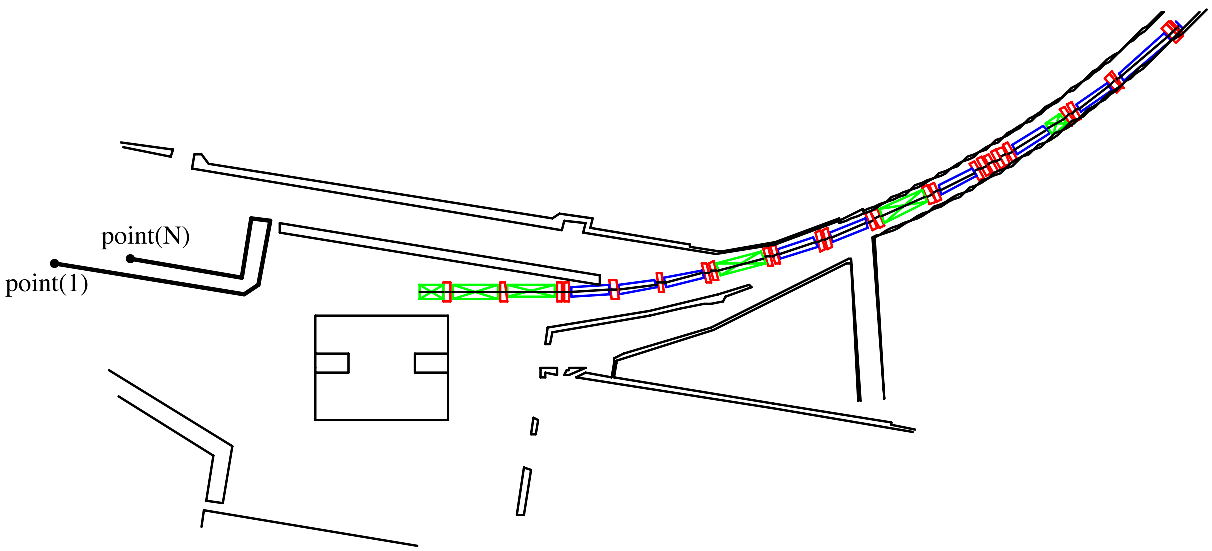
NOTE: Since no range given, the number of elements shown is first 200 of 10323

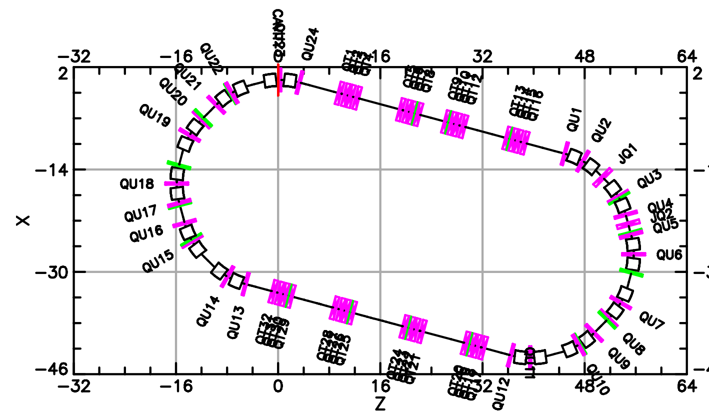
Tao>

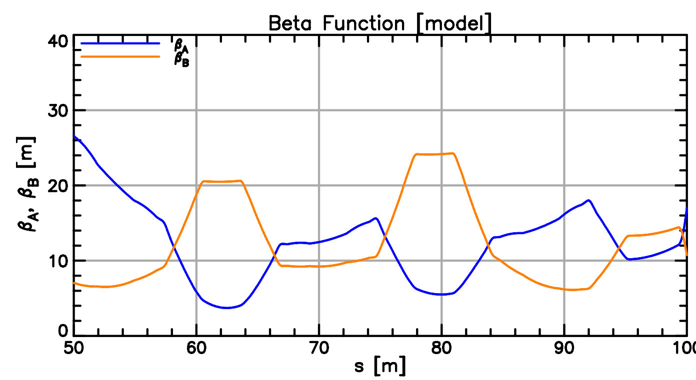
CBETA 4-pass ERL

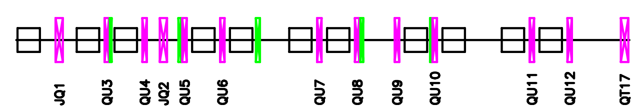
Tao [continued]

Example: Designing or modifying a machine to be/stay within an existing building:



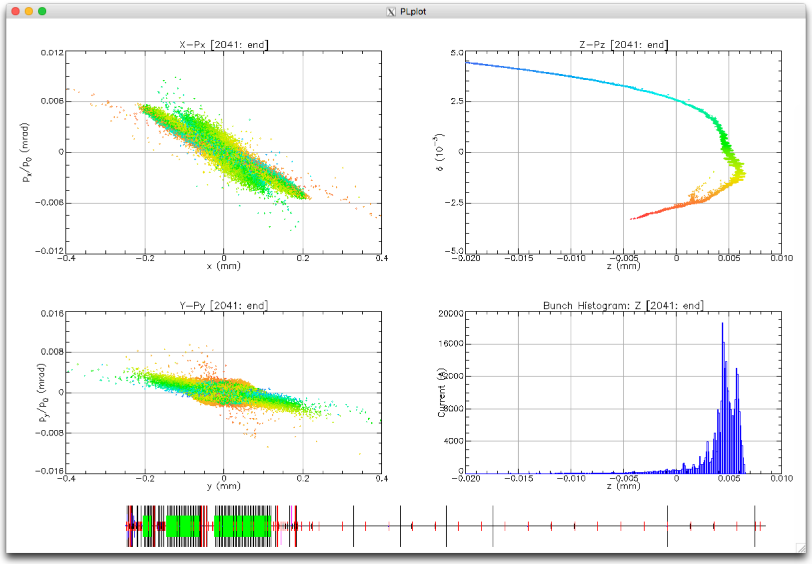


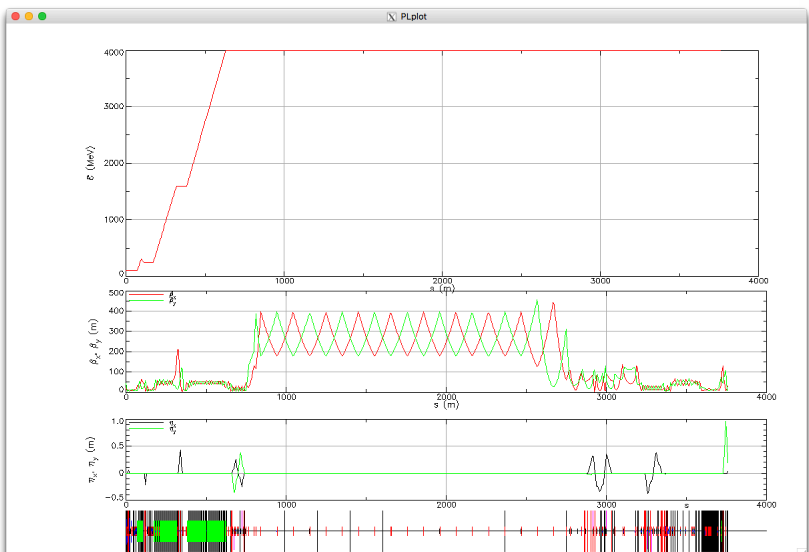




Tao with Bmad gives the flexibility of a library with the convenience of a program.

Tao running the LCLS2 lattice



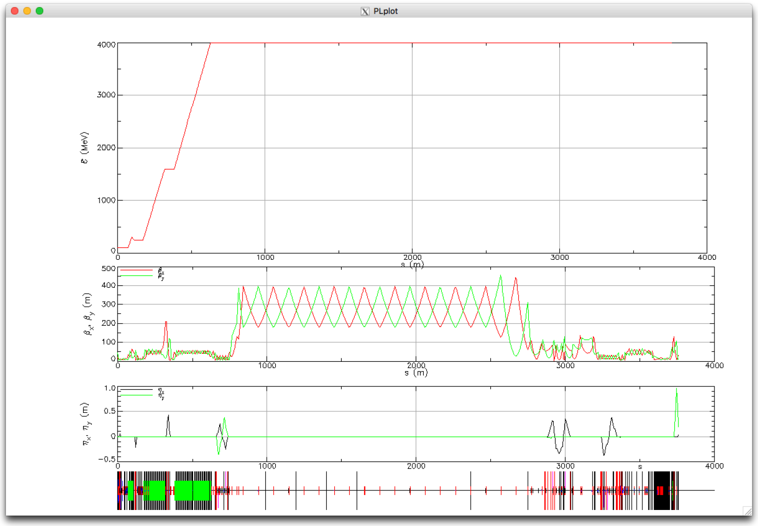


model — tao — 117×64

```
Tao> sho ele 1619
 Element #                1619
 Element Name: CAVL358
 Element Type: "cryo"
 Key: Lcavity
 S_start, S:  631.705741,  632.743485
 Ref_time:  2.110609E-06

 Attribute values [Only non-zero/non-default values shown]:
    1  L                            =  1.0377431E+00 m
    3  RF_FREQUENCY                 =  1.3000000E+09 Hz             RF_WAVELENGTH    =  0.230609581 m
    6  GRADIENT                     =  1.4677427E+07 eV/m
    9  VOLTAGE                      =  1.5231399E+07 Volt
   10  FRINGE_TYPE                  =  Full (4)
   13  SPIN_FRINGE_ON               =  T (1)
   18  AUTOSCALE_AMPLITUDE          =  T (1)
   19  AUTOSCALE_PHASE              =  T (1)
   24  PHI0                         =  2.7777778E-02 rad/2Pi
   31  L_HARD_EDGE                  =  1.1530479E-01 m
   32  FIELD_AUTOSCALE              =  1.0000000E+00
   33  N_CELL                       = 1
   50  DELTA_REF_TIME               =  3.4615385E-09 sec
   51  P0C_START                    =  3.9850000E+09 eV             BETA_START       =  0.999999992
   52  E_TOT_START                  =  3.9850000E+09 eV             DELTA_E          =  1.5000000E+07 eV
   53  P0C                          =  4.0000000E+09 eV             BETA             =  0.999999992
   54  E_TOT                        =  4.0000000E+09 eV             GAMMA            =  7.8278048E+03
   66  NUM_STEPS                    = 104
   67  DS_STEP                      =  1.0000000E-02 m
   76  X1_LIMIT                     =  3.5000000E-02 m
   77  X2_LIMIT                     =  3.5000000E-02 m
   78  Y1_LIMIT                     =  3.5000000E-02 m
   79  Y2_LIMIT                     =  3.5000000E-02 m

         TRACKING_METHOD              =  Bmad_Standard
         MAT6_CALC_METHOD             =  Bmad_Standard
         SPIN_TRACKING_METHOD         =  Tracking
         PTC_INTEGRATION_TYPE         =  Matrix_Kick
         FIELD_CALC                   =  Bmad_Standard
         APERTURE_AT                  =  Exit_End
         OFFSET_MOVES_APERTURE        =  F
         SYMPLECTIFY                  =  F
         FIELD_MASTER                 =  F
         CSR_CALC_ON                  =  T

Slave_status: Free
Lord_status:  Not_a_Lord

Twiss at end of element:
                         A              B            Cbar                        C_mat
  Beta (m)        32.55240066    28.24702055  |  -0.00000000   0.00000000     -0.00000000   0.00000000
  Alpha           -0.72250693     1.01776549  |  -0.00000000  -0.00000000     -0.00000000  -0.00000000
  Gamma (1/m)      0.04675588     0.07207297  |   Gamma_c =    1.00000000       Mode_Flip = F
  Phi (rad)       29.60127042    36.86380039          X              Y              Z
  Eta  (m)        -0.00000000    -0.00000000    -0.00000000    -0.00000000    -1.53091755
  Etap            -0.00000000     0.00000000    -0.00000000     0.00000000    -0.00000000

Orbit:  Electron   State: Alive
         Position[mm] Momentum[mrad]        Spin     |
  X:     -0.00000000    -0.00000000                   | Particle [sec]:      2.11060937E-06  E_tot;  4.0000E+09
  Y:      0.00000000    -0.00000000                   | Part-Ref [sec]:     -4.23516474E-22  PC:     4.0000E+09
  Z:      0.00000000     0.00000000                   | (Ref-Part)*Vel [m]:  1.26967044E-13  Beta:    1.000000

Tao>
```

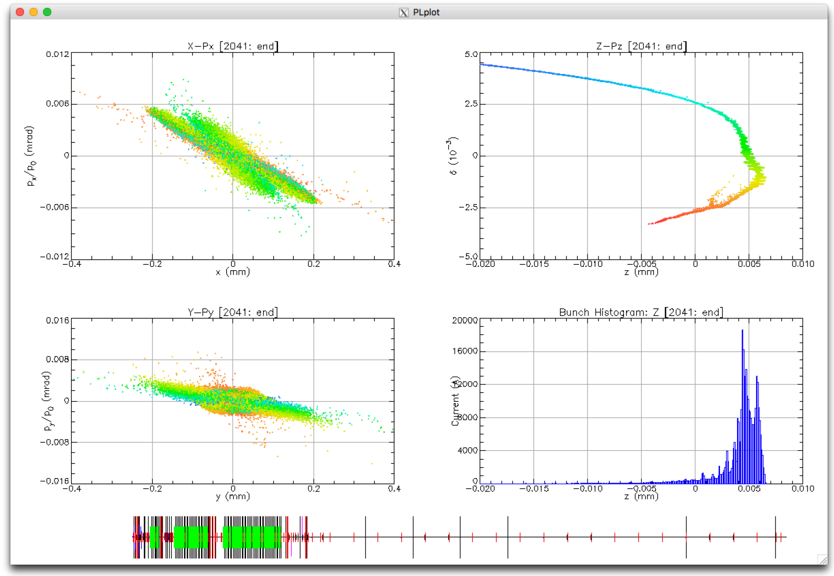
Tao as an online model

Tao Live interactive display



```
Tao> call read_EPICS.tao
     (repeat)
Tao> call write_data.tao
```

CSR, Space Charge tracking



Discover errors

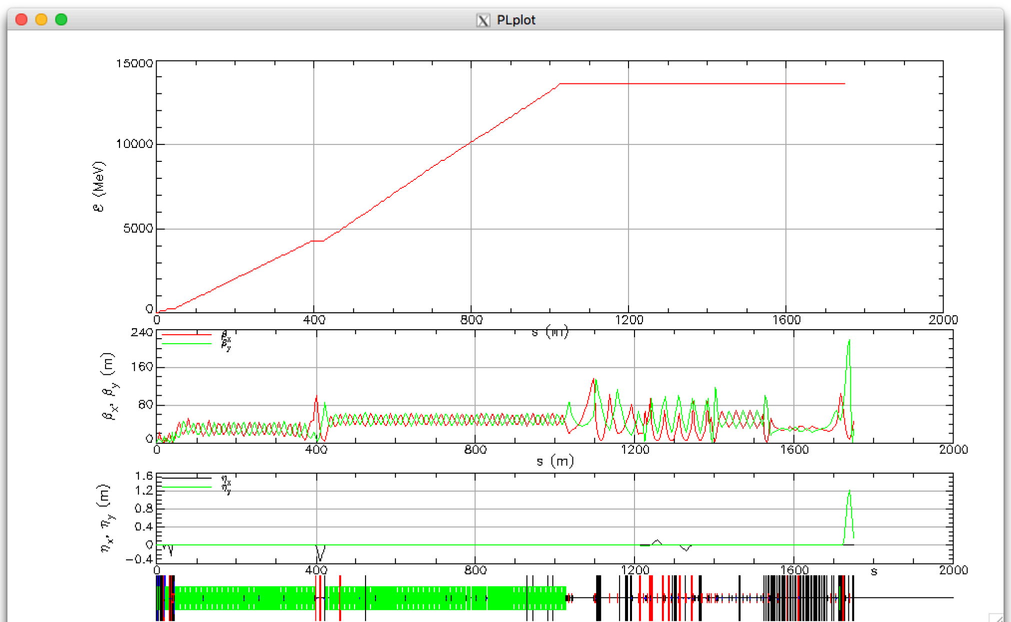
```
Tao> use data bpms
Tao> use variable quad_offsets
Tao> derivative. ! Calculates response of offset quad to BPM
Tao> show derivative

Tao> call read_BPMS_EPICS.tao
Tao> run_optimizer
```

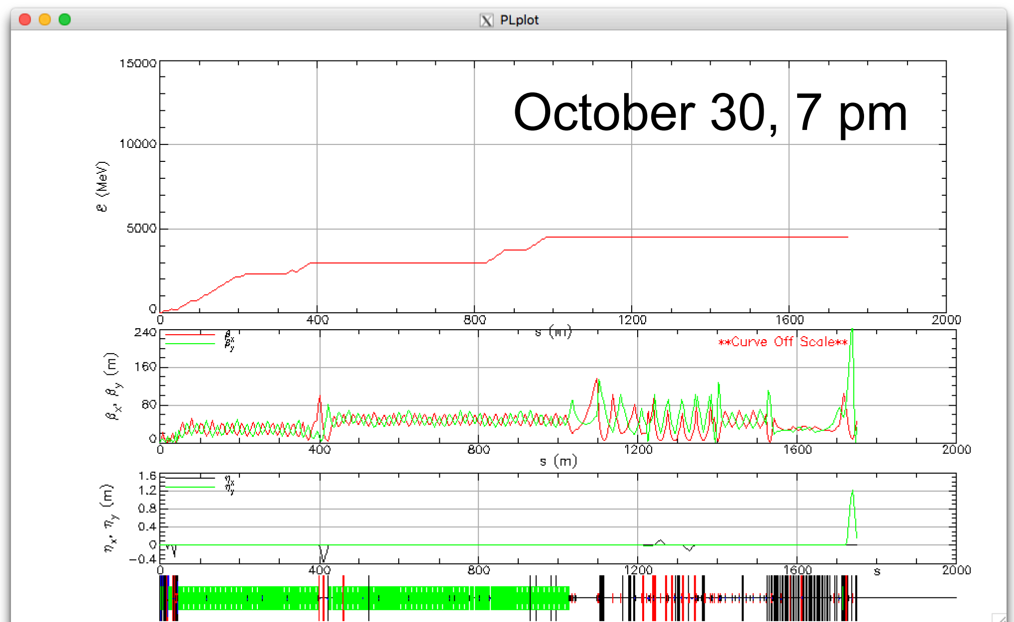
Python wrapper

```
Python >>>
>>> from pytao.tao import Tao
>>> tao = Tao('-lat my_lat.bmad')
>>> tao.command('use var *')
>>> doutput = tao.capture('show top10')
>>> print(output)
```

Bmad online model of LCLS (Current State)

- Complete translation of the MAD lattice, including comments and structure
- Overlays to add functionally for:
  - Klystrons [ENLD, PHAS, STAT], including feedback klystrons
  - Bunch compressors [angle]
  - Linac overall [phase, fudge]
- Python scripts read EPICS PV values, write Bmad compatible statements
- Tao script for ‘LEMing’ based on energy measurement PVs






```
Design lattice in:
/u/ad/cmayes/LCLS/model
```

```
Tao> call scripts/live_update.tao
```

Summary Bmad advantages

SLAC

- State-of-the-art tracking methods, fast and slow
- Arbitrary trajectory 1-D CSR model with shielding
- Medium-High energy space charge model
- New: 3D Space Charge
- Field maps for any element, can overlap other elements
- Superposition: Greatly simplifies lattice layout, bookkeeping
- Patch element: Enables arbitrary arrangement of magnets
- Controller elements: define arbitrary knobs
- Continuous beam chamber walls, masks
- Forking: Multiple connected lines
- Reads MAD, XSIF, SAD lattice formats
- Translation routines to Astra, OPAL, MAD, XSIF, SAD, …
- Spin tracking
- X-ray tracking
- Dark current tracking
- Software toolkit: Maximum flexibility for custom programs

Summary Tao advantages

- Design tool
- Fast online optics calculation
- Multiple optimization methods
- Bunch tracking with CSR, Space charge
- Response of anything to anything: effect of quad misalignments, etc.
- Wave analysis: discover isolated orbit and focusing errors.
- Built-in plotting
- Customizable via hook routines, e.g. interface with a control system
- Python interface for automation, advanced GUIs, also interface with control systems

SLAC FEL Lightsource Modeling

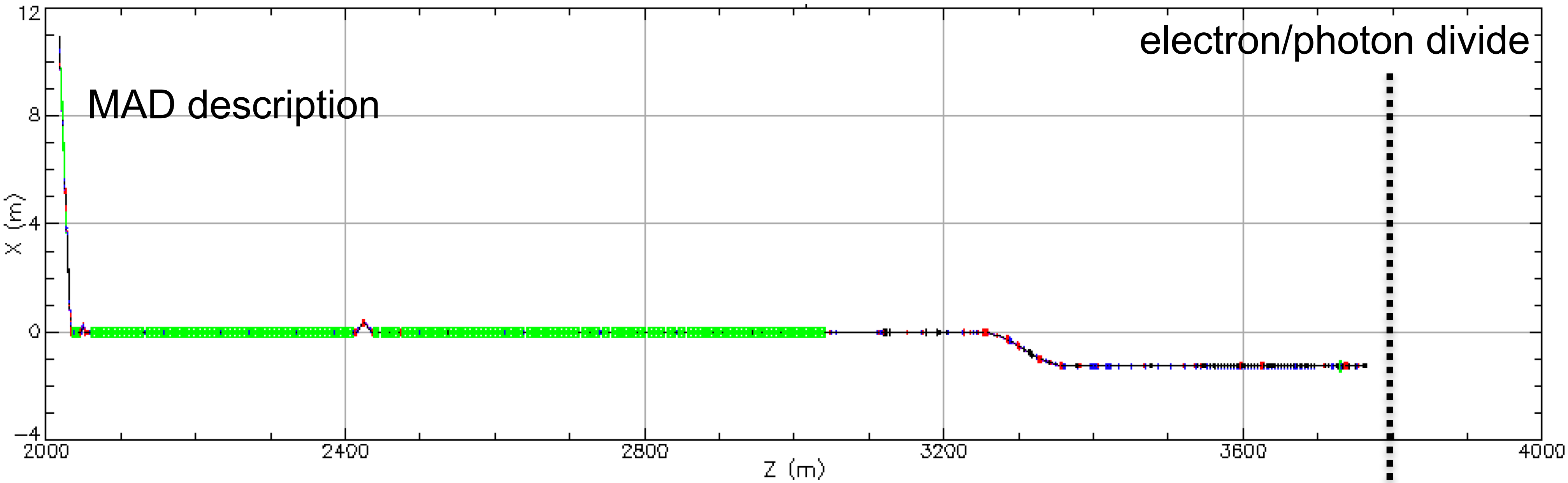
Goals:

- Maximize FEL performance.
- Minimize downtime (rapid/automated tuneup, diagnose problems).
- Enable advanced beam dynamics studies.

Existing modeling problems:

- LCLS: Measured optics do not match existing model, no collective effects
- LCLS-II: Collective effects and their impact on FEL performance are not fully understood
- LCLS-II-HE: Space charge limits the injected emittance; What bunch qualities are really needed at injection?

Current state of modeling



Simulation codes

Astra

ImpactT

ImpactZ

elegant

SLAC MAD8

SLAC Matlab MAD

Genesis

?

SLAC FEL Lightsource Modeling

Proposal:

- Construct realistic start-to-end models of the electron machines (LCLS, LCLS-II, LCLS-II-HE).
- Develop a simulation pipeline for tracking electrons from the cathode through X-ray production.
- Bridge this to X-ray transport simulation pipeline.

Benefits

A realistic start-to-end model and simulation framework will enable holistic optimization, understanding, and performance prediction of SLAC's lightsources.

- For the existing machines (LCLS, soon LCLS-II), this would be used as an operational tool to understand and tune-up the machine.
- For future machines (LCLS-II-HE and beyond) it would be used to understand requirements for various components (e.g. electron gun, injector) and sensitivities.
- Subtle beam physics effects that limit ultimate performance would be identified early on.
- When bridged with X-ray transport, this would be used to predict the success of individualized experiments.
- The model could be optimized as a whole (correlate experiment performance with accelerator parameters), or be used to train machine learning algorithms for online control.

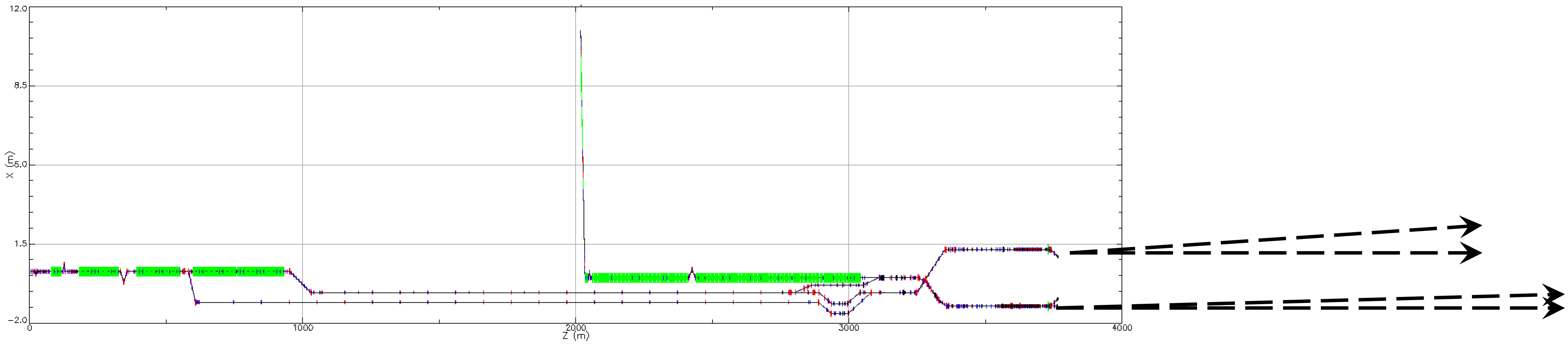
Plan

SLAC

Most of the simulation codes and techniques exist, but are not bridged. We will:

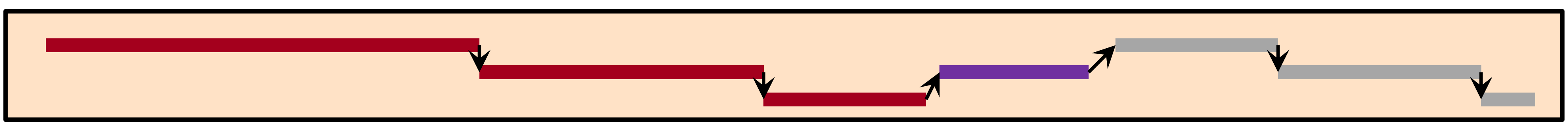
- Identify the relevant codes and human expertise for each of the parts of the machine (injector, Linac, beam transport, FEL light production, X-ray optics, X-ray final focus, detector)
    - Avoid re-inventing the wheel: Use existing, well-developed codes as much as possible (e.g. Impact, Bmad, Genesis)
- Quantify and experimentally validate various beam effects (e.g. space charge, coherent synchrotron radiation, wakefields)
- Tune models to match machine measurements (e.g. alignment, field errors)
- Develop a software wrapper to enable ‘push button’ start-to-end simulation runs usable by both experts and students.
- Collaborate with the X-ray optics group (which is pursuing a parallel effort) for truly start-to-end simulations.

Future: LCLS-I, LCLS-II, LCLS-II-HE, …



Simulation tools

Custom simulation pipeline



Start-to-End simulation