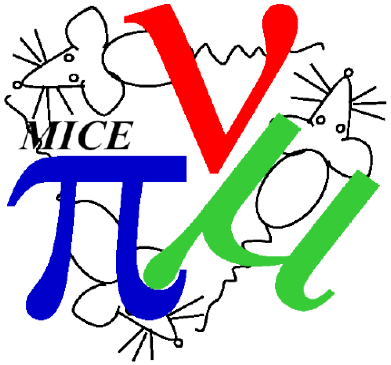
MAUS – Analysis Issues



Chris Rogers,
ASTeC,
Rutherford Appleton Laboratory

MAUS Items for Analysis

- Analysis group are interested in certain items in MAUS
  - Data structure + representation
    - Interface to analysis
  - Routines for generating Monte Carlo data sets
    - Used for doing truth vs recon studies
- Analysis group owns certain items in MAUS
  - Global reconstruction
  - Cooling channel geometry
  - Online tools for tuning the accelerator
- Go through these one by one

Data Representation

- Data currently represented in json
  - No way to enforce any structure
  - Developers can e.g. change the data structure during reconstruction
  - No way to enforce conventions
    - Like “all Space Points should have (x,y,z,time) field”
  - Some concerns over file size, access speed
    - Can probably be addressed using native json, e.g. use some compression routine
  - Naturally leads away from good practice – encapsulation, etc
- Propose using a ROOT data tree
  - More conventional
  - Maintain json functionality

Data Structure

- Active discussion of how data is structured in MAUS
- Do we split into particle events after unpacking?
- Do we split into particle events after detector reconstructions?
- Do we group by reconstruction “stage“ or by detector
- Needs to be closed out and implemented by March run
  - Looks potentially tight
- Non-controversial
  - Add some metadata to the structure e.g. what run, what configuration, etc...

Batch Production

- Propose a batch production job for MC and Reconstruction
  - After a stage of settling analysis users won't be interested in details of this or that reconstruction routine
  - Nice to have “official” recon data sets for analysis
- Will be done on GRID
  - Automatically trigger as data is taken
  - Use a reference version which may not be the latest – chosen by analysis group
  - Rerun over data sets for new code version/calibration/etc can be requested by analysis group
- Needs (by mid-March):
  - GRID setup – okay
  - Input beam (next slide) – unlikely, workaround possible
  - Geometry import – at risk
  - Digitisation of TOF and other detectors - okay
  - ROOT IO and data structure – at risk

Input Beam (g4bl)

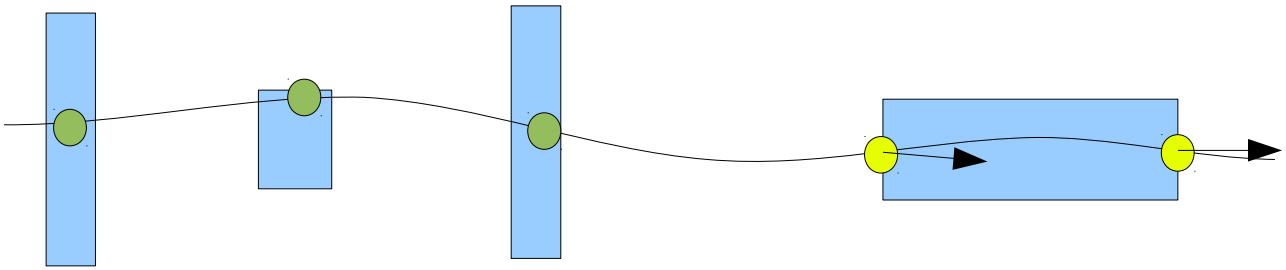
- I think we use G4BL for input beam to batch production
  - Need reasonable rate and time distribution
    - For multiparticle effects (pile up, dead time)
    - For tracker dead time model
  - Extract rate from GVA1
    - Fit to binomial distribution?
  - Extract time distribution in each spill from fit to TOF0 t
    - Fit to some exponential distribution? Saw tooth?
  - Means we need to do reconstruction job and then use reconstructed GVA1 and TOF0 to do the Monte Carlo
    - Circular dependency is unpleasant
      - Use recon as input to MC which is used to understand recon... urg...
    - Is there a better way?

Geometry Responsibility

- We have a set of routines for pulling geometry from a reduced CAD model of MICE
- It needs someone to check that the geometry physics is correct
  - Detectors are responsible for checking that their geometry is correct
  - Beamline expert is responsible for checking geometry and fields in beamline is correct
  - Analysis group is responsible for checking cooling channel geometry and fields are correct
- Geometry release cycle goes like
  - CAD release from Jason Tarrant
  - Converted into Geant4 readable and uploaded to ConfigDB
  - Checked by experts
  - Tagged as a release of new geometry
- First attempt will follow survey being taken right now

Global Recon

- We have some set of measured space points (TOF), tracks (tracker), Light yields (EMR, KL, Ckov) with some set of errors
- How do we tie these things together?
- Two somewhat separate things:
  - Track fitting
  - PID



Basic specification

- We seek to get the following variables
  - x, y, z, time
  - px, py, pz, energy
  - Preferred pid (gives charge, mass, etc)
  - On user-defined planes – typically spacial planes, probably not z-planes
    - Engineering geometry has ***x*** as the beam axis
- We seek errors
  - Covariance matrix of errors in (x, y, time, px, py, energy)
    - E.g. say we know x exactly but px we have some error
    - Propagate that measurement and we find we introduce uncertainty in x
    - In fact there is a correlation between the x-px uncertainty
    - Encode as a “covariance matrix”
  - p(electron), p(muon), p(pion)? sigma(mass)? Not sure how to write down error in pid

Track/Error propagation

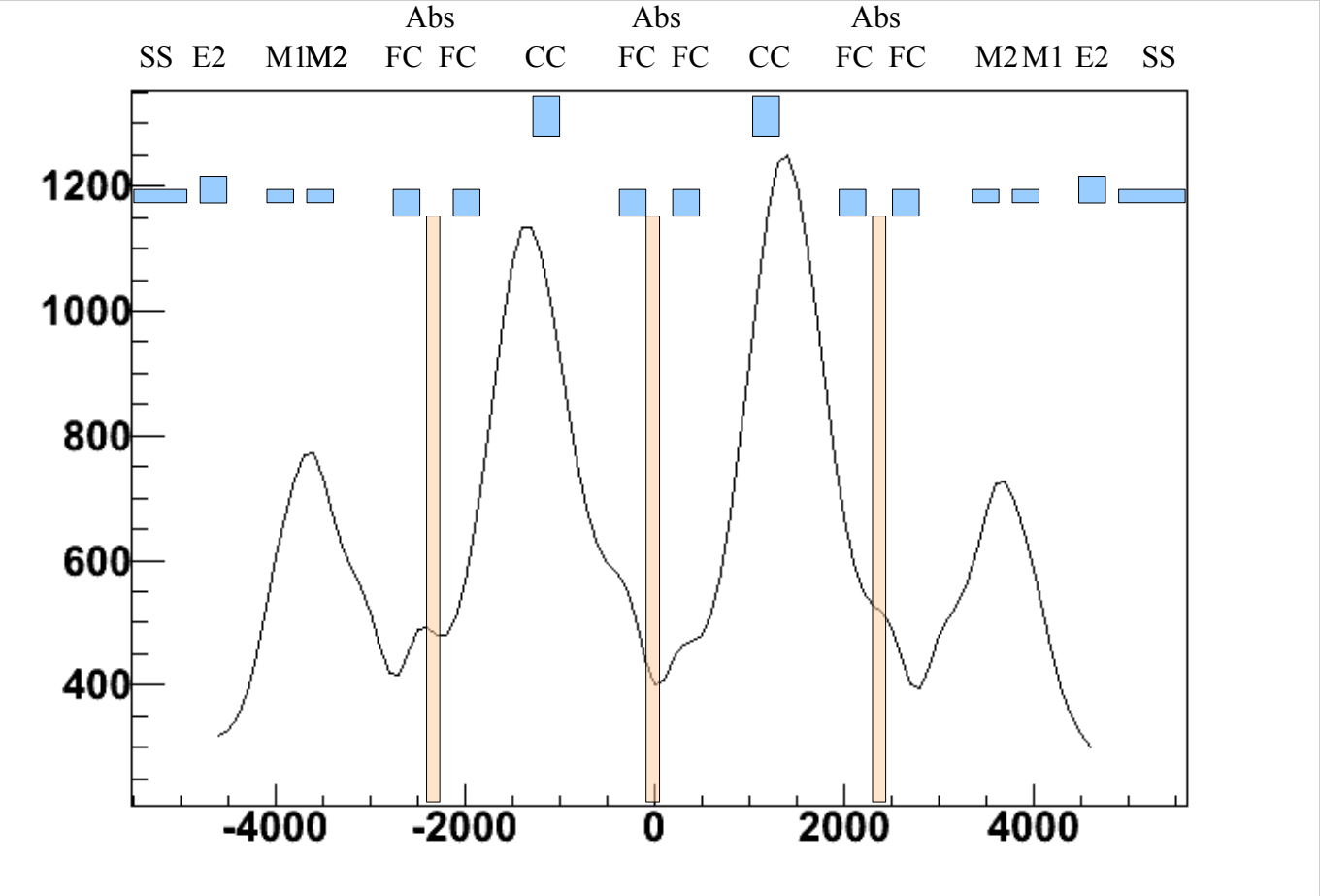
- We can already do Runge Kutta through arbitrary field maps, either using G4 or custom tracking routines
- We can already calculate transfer matrices through arbitrary field maps
  - Numerical differentiation of the tracking
    - Generate a few different tracks
    - Do $u_{out} = M\, u_{in}$
    - We can use G4 to calculate u_{out} from u_{in}
    - We can solve as a simultaneous equation for **M**
  - Second order correction implemented and works
  - Direct integration not implemented but could be
- All implemented in Optics routines
  - Needs significant clean up and testing

Next Steps

- Specification document is in progress
  - Aim to have first draft ready next week at software workshop
- Definition of data structure
- Definition of class structure
  - We already have some optics infrastructure in place, currently in a tidy-up phase
- Seek comments early and often
  - I am a novice – analysis experts in MICE have done it all before

Optics Tool

- Request was made for a light weight tool to evolve beam envelopes for use in control room
- Propose following tool
  - (powerpoint prototype of a GUI)
  - Allows to define a set of magnet currents and input beam
  - Then plot resultant beam envelope, momenta, etc down the accelerator
  - Geometry shown is Stage 6 geometry
  - In principle we should be able to handle any geometry e.g. would hope to include Q4-9, etc
- Seek comments



Beam

Z (mm) -2750

definition penn

beta (mm) 420

alpha 0

em. (mm) 6

Beamline

E1 (A) 149.05

M1 (A) 149.05

SS (A) 222.01

M2 (A) 222.01

E2 (A) 109.12

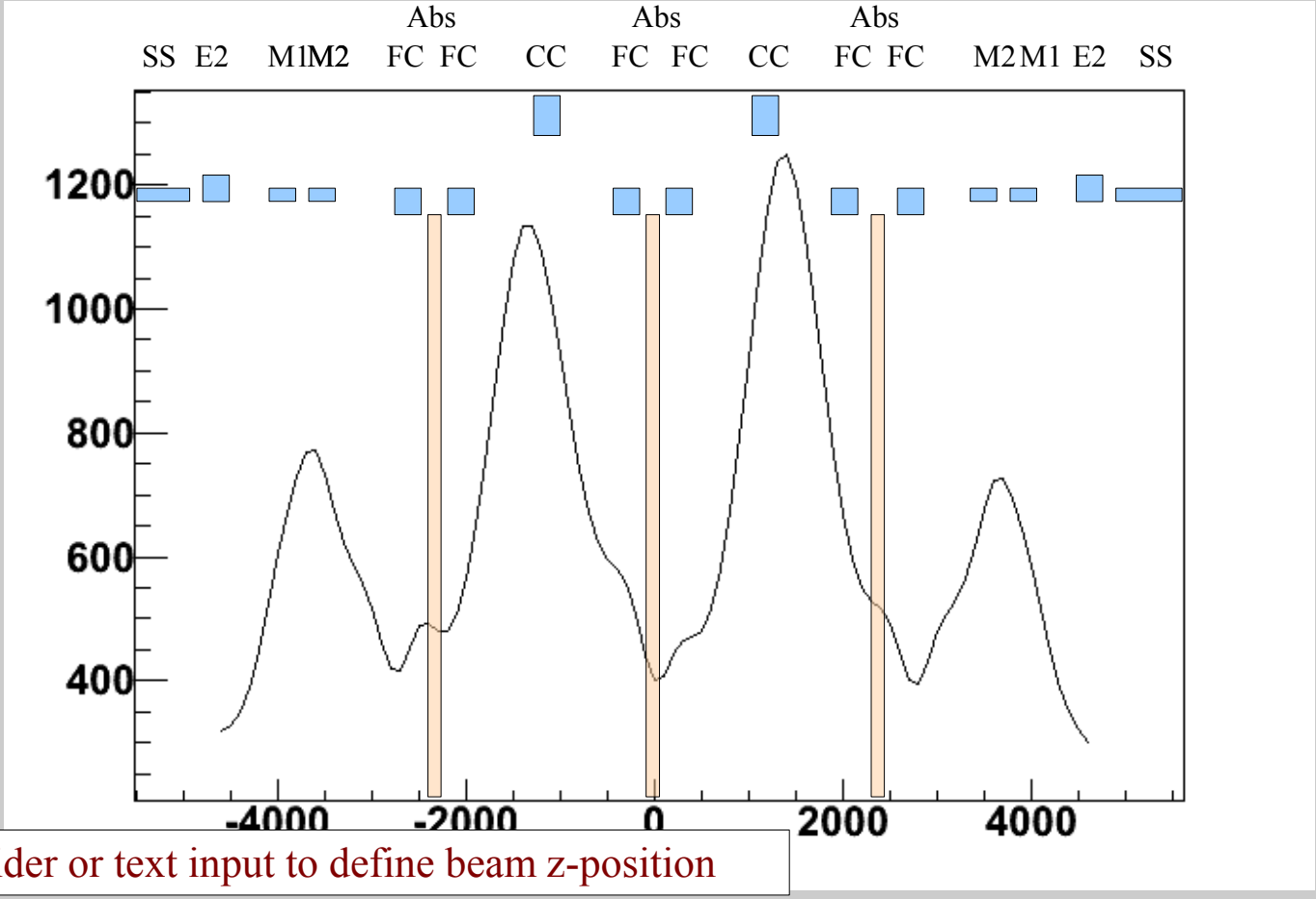
FC (A) 109.12

CC (A) 109.12

Plot

ordinate beta_perp

Show optical elements



Slider or text input to define beam z-position

Beam

Z (mm) -2750 definition penn

beta (mm) 420

alpha 0

em. (mm) 6

Beamline

E1 (A) 149.05 M1 (A) 149.05

SS (A) 222.01 M2 (A) 222.01

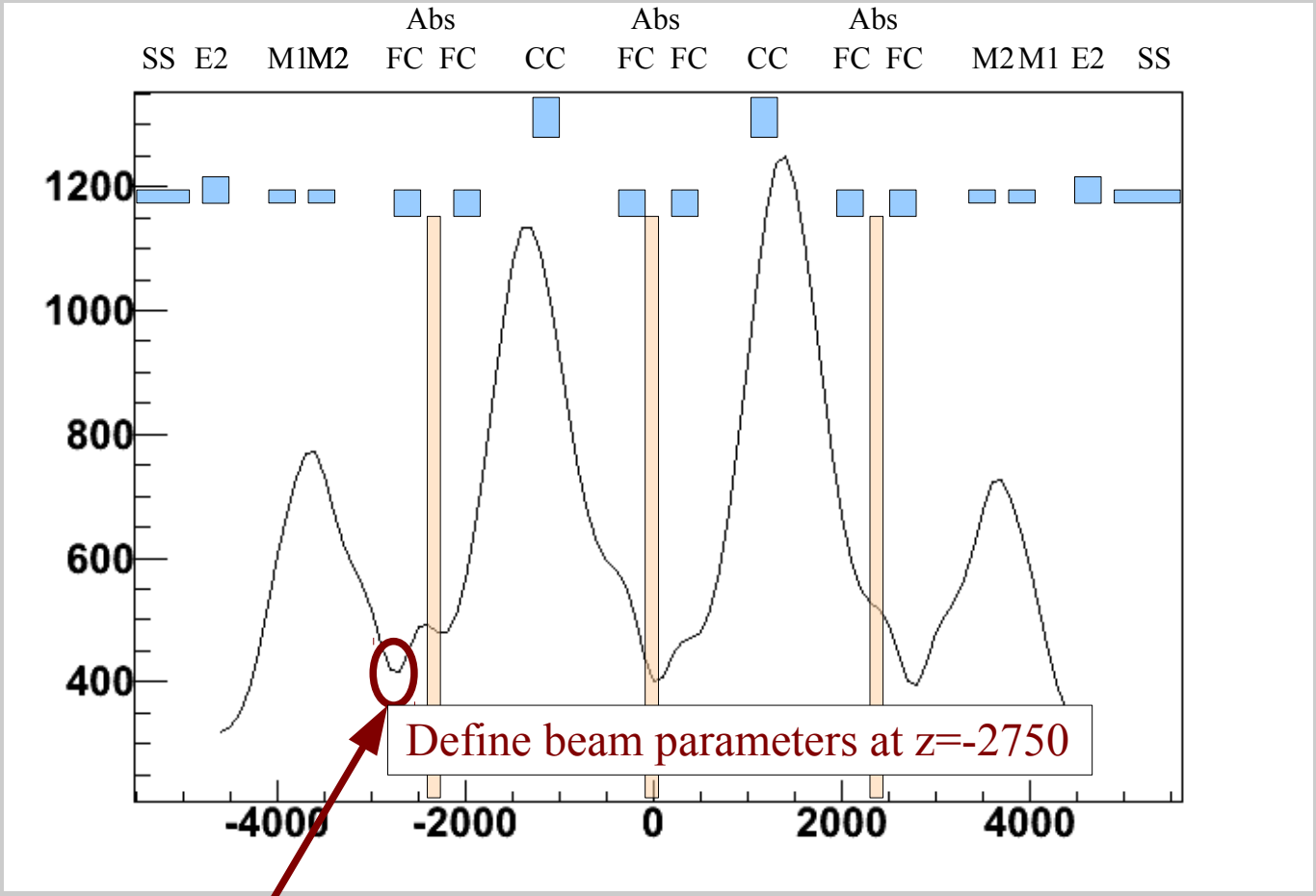
E2 (A) 109.12 FC (A) 109.12

CC (A) 109.12

Plot

ordinate beta_perp

Show optical elements



Beam

Z (mm) -2750

definition penn

beta (mm) 420

alpha 0

em. (mm) 6

Beamline

E1 (A) 149.05

M1 (A) 149.05

SS (A) 222.01

M2 (A) 222.01

E2 (A) 109.12

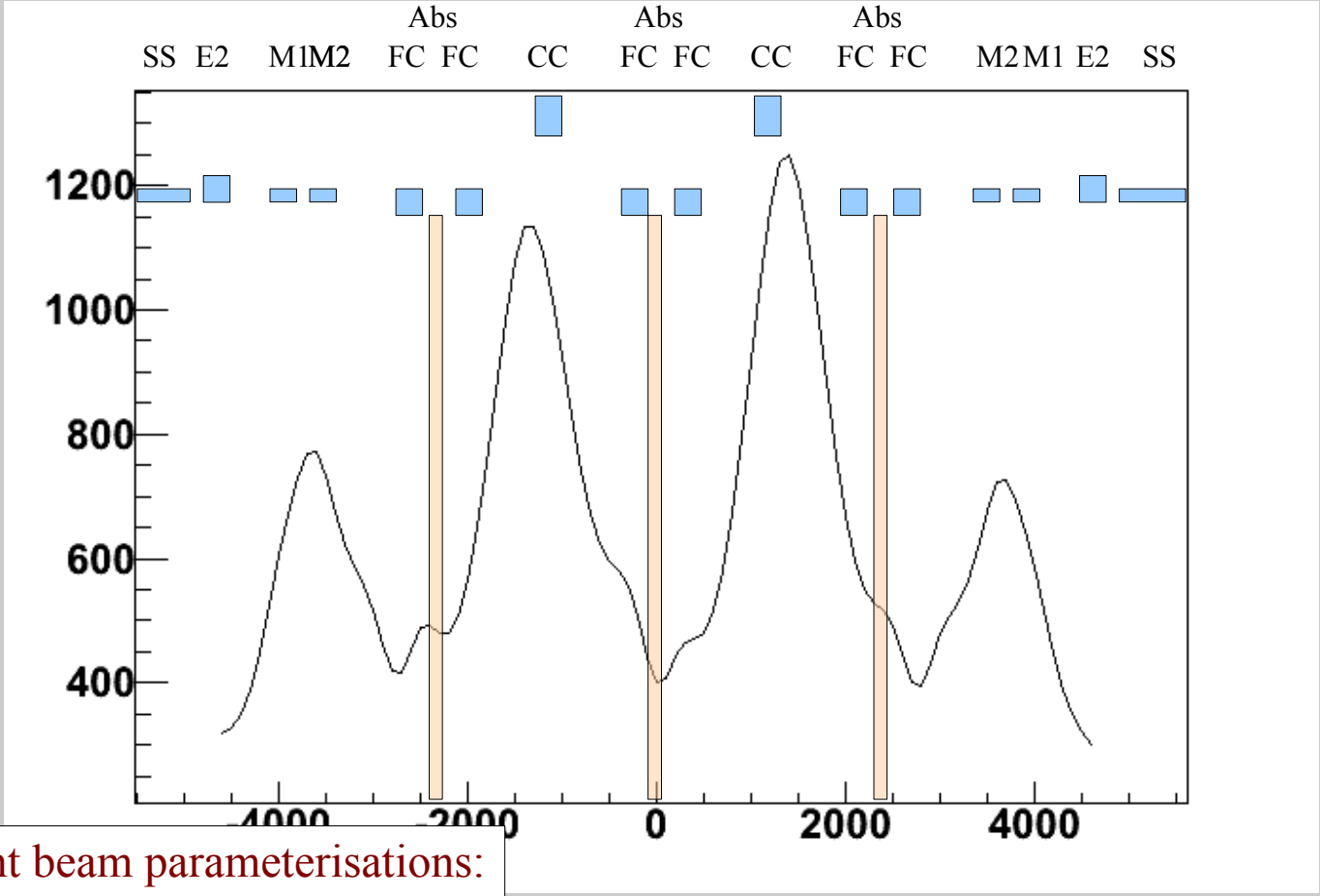
FC (A) 109.12

CC (A) 109.12

Plot

ordinate beta_perp

Show optical elements



Different beam parameterisations:
penn
twiss

Z (mm) -7000 definitio twiss

beta_x (mm) 2700 beta_y (mm) 350

alpha_x 0 alpha_y 0

em_x (mm) 2 em_y (mm) 4

Beamline

E1 (A) 149.05 M1 (A) 149.05

SS (A) 222.01 M2 (A) 222.01

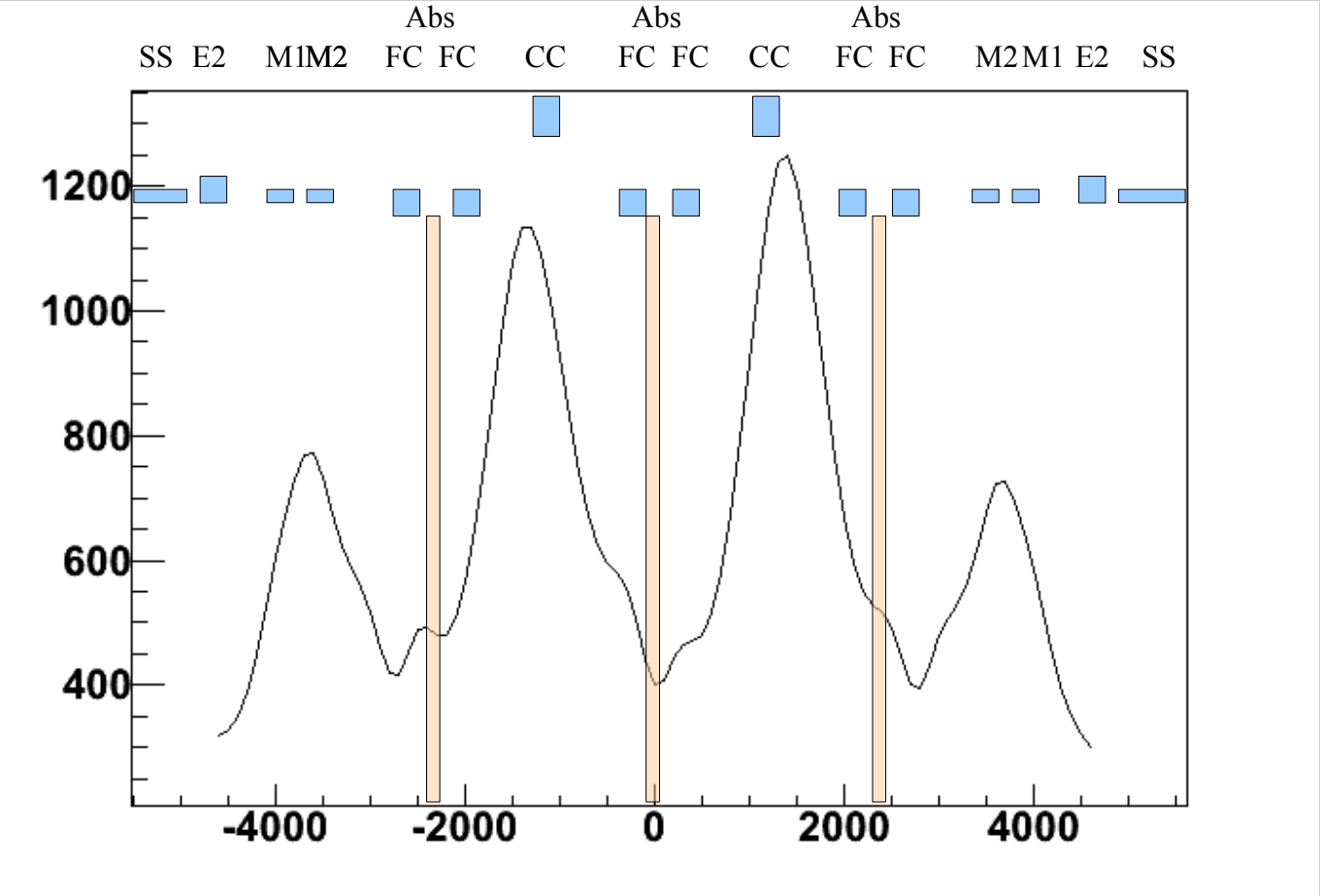
E2 (A) 109.12 FC (A) 109.12

CC (A) 109.12

Plot

ordinate beta_perp

Show optical elements



Beam

Z (mm)

beta (mm)

alpha

em. (mm) 6

Control magnet currents
•Should use currents like the ones typed into controls PC

Beamline

E1 (A) 149.05 M1 (A) 149.05

SS (A) 222.01 M2 (A) 222.01

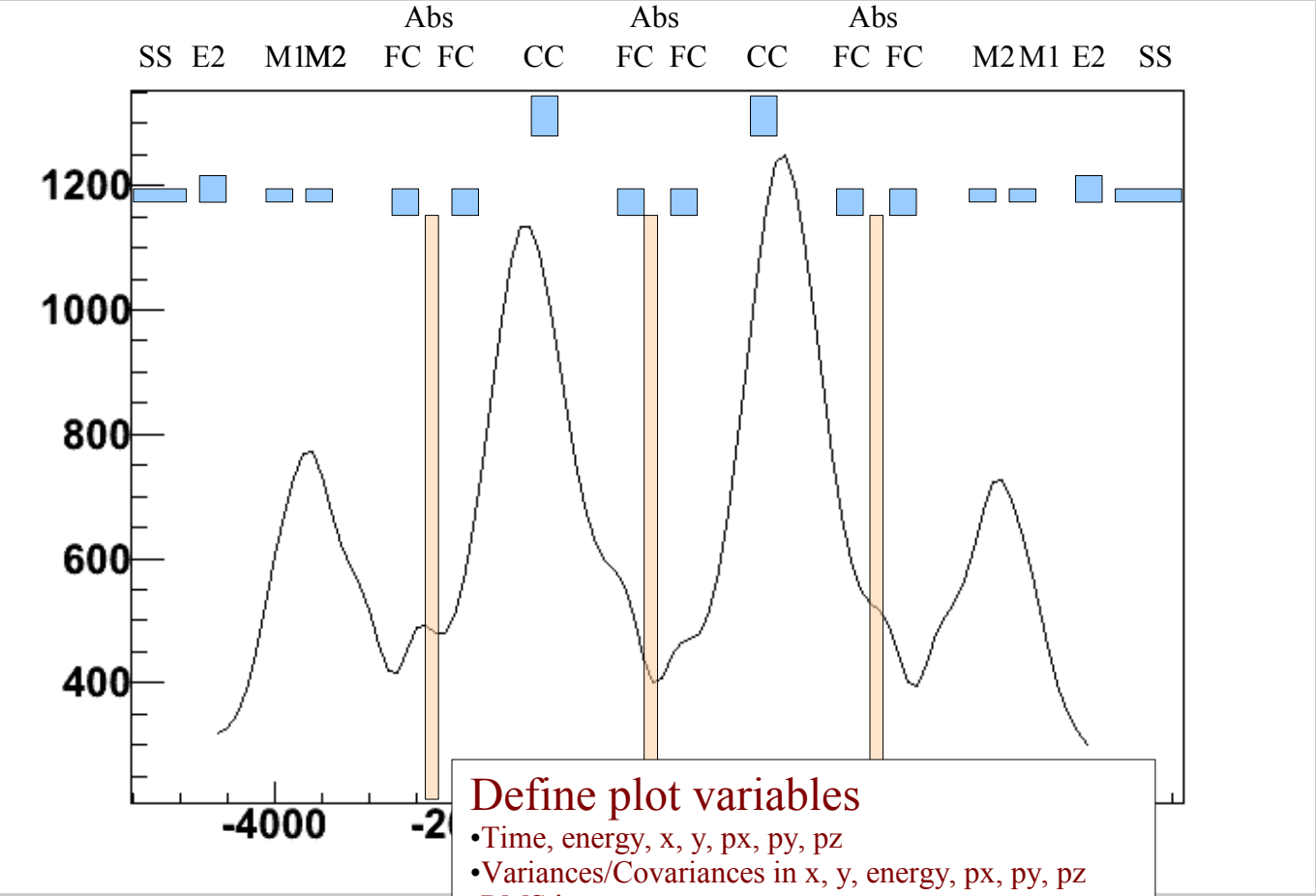
E2 (A) 109.12 FC (A) 109.12

CC (A) 109.12

Plot

ordinate beta_perp

Show optical elements



Define plot variables

- Time, energy, x, y, px, py, pz
- Variances/Covariances in x, y, energy, px, py, pz
- RMS in x, y, energy, px, py, pz
- Beta function in x, y, transverse 4D
- Alpha function in x, y, transverse 4D
- Gamma function in x, y, transverse 4D
- Emittance in x, y, transverse 4D
- Kinetic angular momentum
- Amplitude momentum correlation
- Dispersion in x, y
- Dispersion prime in x, y

Beam

Z (mm) -2750

definition penn

beta (mm) 420

alpha 0

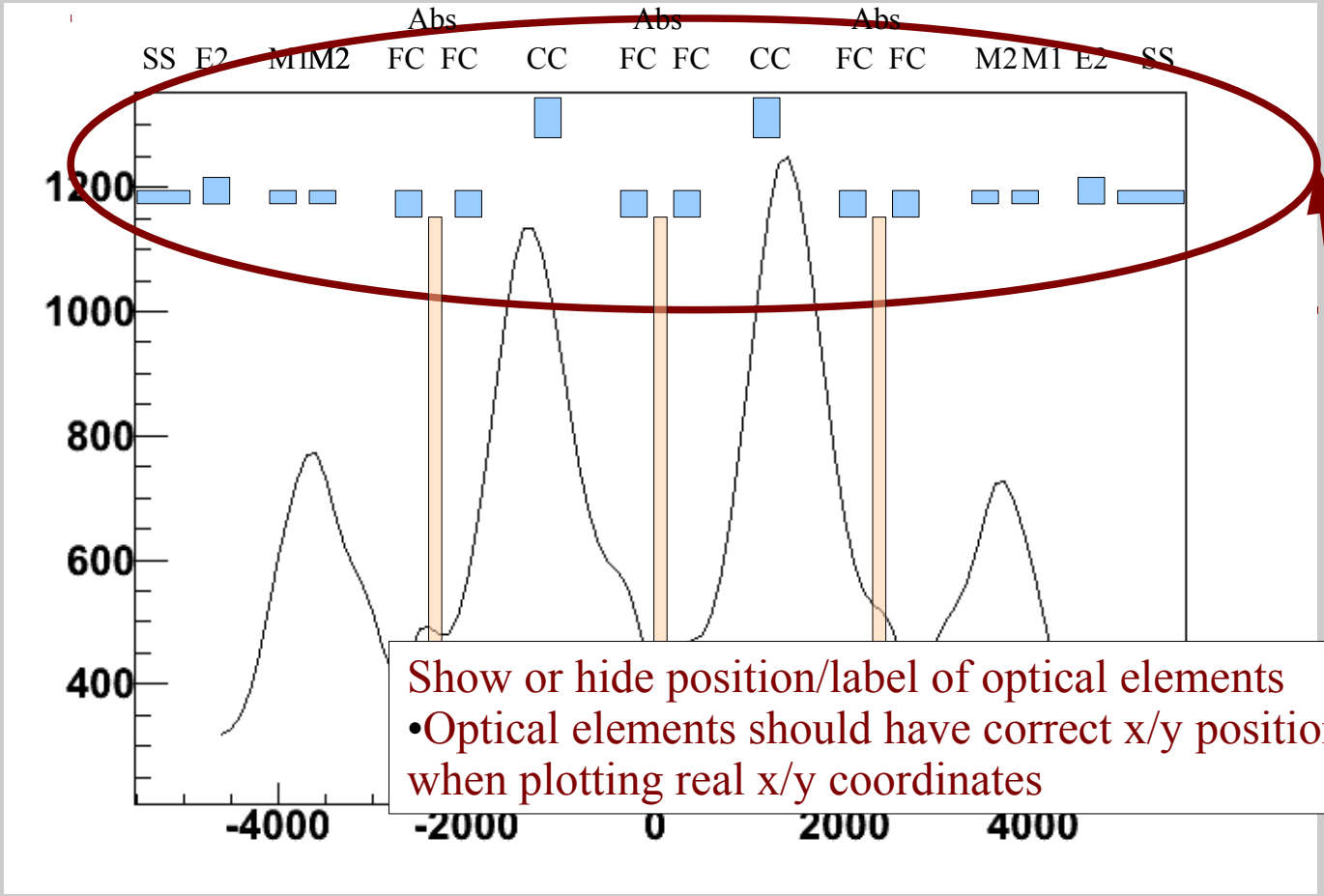
em. (mm) 6

CC (A) 109.12

Plot

ordinate beta_perp

Show optical elements



Show or hide position/label of optical elements
•Optical elements should have correct x/y position when plotting real x/y coordinates

Beam

Z (mm) -2750

definition penn

beta (mm) 420

alpha 0

em. (mm) 6

Beamline

E1 (A) 149.05

M1 (A) 149.05

SS (A) 222.01

M2 (A) 222.01

E2 (A) 109.12

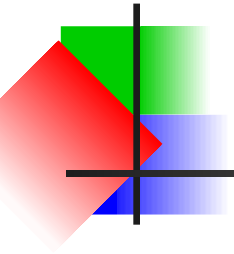
FC (A) 109.12

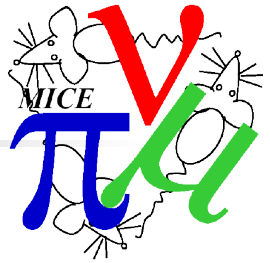
CC (A) 109.12

Plot

ordinate beta perp

Show optical elements





FIN