Data management and analysis for beam dynamic simulation

Denis Zyuzin

Institute for Nuclear Physics
Forschungszentrum Juelich

September 25, 2013

Motivation

We've been using COSY Infinity during the last 2 years and these are some pluses and minuses of the program.

Advantages:

- Supports both electric and magnetic elements
- Simulates spin motion
- Can do very fast symplectic tracking for million turns
- Coincides with analytical estimations
- Fringe fields calculation

Difficulties:

- Not easy to use, have to read manual
- Some features not included to the user manual
- Many lattices written in MAD incompatible with COSY Infinity, converter is needed
- Non informative error messages (like "PRODUCING TRACEBACK BY DELIBERATE ILLEGAL OPERATION SQRT(-1.D0)")

Motivation

We've been using COSY Infinity during the last 2 years and these are some pluses and minuses of the program.

Advantages:

- Supports both electric and magnetic elements
- Simulates spin motion
- Can do very fast symplectic tracking for million turns
- Coincides with analytical estimations
- Fringe fields calculation

Difficulties:

- Not easy to use, have to read manual
- Some features not included to the user manual
- Many lattices written in MAD incompatible with COSY Infinity, converter is needed
- Non informative error messages (like "PRODUCING TRACEBACK BY DELIBERATE ILLEGAL OPERATION SQRT(-1.D0)")

Motivation

Typical workflow for tracking particles with COSY Infinity:

1. Create source fox-file
2. Try to run, fix syntax errors, run again

Motivation

Typical workflow for tracking particles with COSY Infinity:

1. Create source fox-file
2. Try to run, fix syntax errors, run again
3. When tracking is finished, analyze results and save it somewhere

Motivation

Typical workflow for tracking particles with COSY Infinity:

1. Create source fox-file
2. Try to run, fix syntax errors, run again
3. When tracking is finished, analyze results and save it somewhere

After two years of using the program we have 43766 files with tracking results named 1.out, 1.ray, 2299.ray, output.spin.98.dat, etc. and it's impossible neither find something there nor reproduce the results (because unable to find source fox-file used for tracking a year ago).

Motivation

Typical workflow for tracking particles with COSY Infinity:

1. Create source fox-file
2. Try to run, fix syntax errors, run again
3. When tracking is finished, analyze results and save it somewhere

After two years of using the program we have 43766 files with tracking results named 1.out, 1.ray, 2299.ray, output.spin.98.dat, etc. and it's impossible neither find something there nor reproduce the results (because unable to find source fox-file used for tracking a year ago).

A new tool

All steps to work with COSY Infinity can be done automatically:

- Compile COSY Infinity binary from Fortran sources
- Compile beam dynamics package from cosy.fox file

A new tool

All steps to work with COSY Infinity can be done automatically:

- Compile COSY Infinity binary from Fortran sources
- Compile beam dynamics package from cosy.fox file
- Generate fox files for analyzing lattice (tunes/chromaticities calculation), particle tracking (with different initial distributions)

A new tool

All steps to work with COSY Infinity can be done automatically:

- Compile COSY Infinity binary from Fortran sources
- Compile beam dynamics package from cosy.fox file
- Generate fox files for analyzing lattice (tunes/chromaticities calculation), particle tracking (with different initial distributions)
- Analyzing results: plotting tracking results, comparing different initial parameters

A new tool

All steps to work with COSY Infinity can be done automatically:

- Compile COSY Infinity binary from Fortran sources
- Compile beam dynamics package from cosy.fox file
- Generate fox files for analyzing lattice (tunes/chromaticities calculation), particle tracking (with different initial distributions)
- Analyzing results: plotting tracking results, comparing different initial parameters
- Storing all calculated data in the database (if something was calculated once, it will be not recalculated, but fetched from the database)

A new tool

All steps to work with COSY Infinity can be done automatically:

- Compile COSY Infinity binary from Fortran sources
- Compile beam dynamics package from cosy.fox file
- Generate fox files for analyzing lattice (tunes/chromaticities calculation), particle tracking (with different initial distributions)
- Analyzing results: plotting tracking results, comparing different initial parameters
- Storing all calculated data in the database (if something was calculated once, it will be not recalculated, but fetched from the database)

Features

- Simple web-interface, should work on every browser
- You don't need to read the manual and to write fox-code
- Supports all elements and features of COSY Infinity, like electric/magnetic elements, misalignments, fringe fields etc.
- Calculate tunes/chromaticities in one click
- Plot dependencies on system parameters (e.g. plot Q_x depending on quadrupole strength)
- Track with different initial distributions (exact points/Gaussian distribution/Uniform random distribution)
- Plot tracking results
- Able to run "raw" fox-code
- Built-in parallelization

A new tool

Workflow to use system:

1. Create lattice
2. Optimize lattice parameters

A new tool

Workflow to use system:

1. Create lattice
2. Optimize lattice parameters
3. Save lattice and track particles

A new tool

Workflow to use system:

1. Create lattice
2. Optimize lattice parameters
3. Save lattice and track particles
4. Analyze tracking results

A new tool

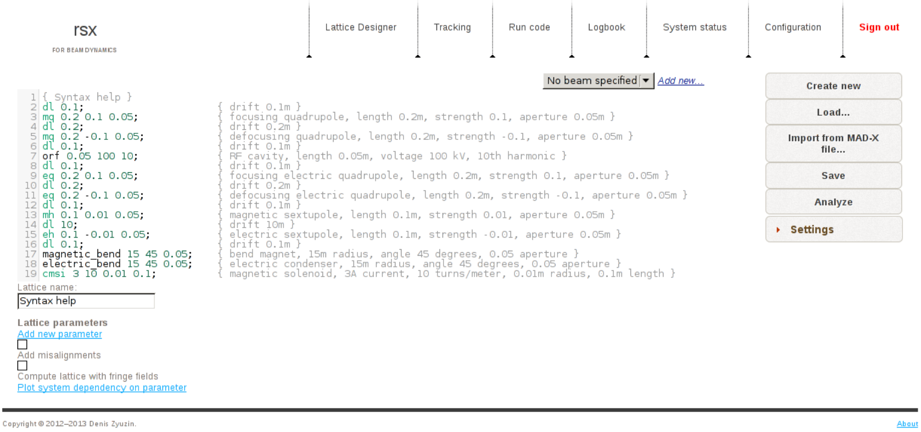
Workflow to use system:

1. Create lattice
2. Optimize lattice parameters
3. Save lattice and track particles
4. Analyze tracking results

Demo

void

Main page


rsx
FOR BEAM DYNAMICS
Lattice Designer
Tracking
Run code
Logbook
System status
Configuration
Sign out
No beam specified
Add new...
Create new
Load...
Import from MAD-X file...
Save
Analyze
Settings
{ Syntax help }
dl 0.1;
mq 0.2 0.1 0.05;
dl 0.2;
mq 0.2 -0.1 0.05;
dl 0.1;
orf 0.05 100 10;
dl 0.1;
eq 0.2 0.1 0.05;
dl 0.2;
eq 0.2 -0.1 0.05;
dl 0.1;
mh 0.1 0.01 0.05;
dl 10;
eh 0.1 -0.01 0.05;
dl 0.1;
magnetic_bend 15 45 0.05;
electric_bend 15 45 0.05;
cmsi 3 10 0.01 0.1;
{ drift 0.1m }
{ focusing quadrupole, length 0.2m, strength 0.1, aperture 0.05m }
{ drift 0.2m }
{ defocusing quadrupole, length 0.2m, strength -0.1, aperture 0.05m }
{ drift 0.1m }
{ RF cavity, length 0.05m, voltage 100 kV, 10th harmonic }
{ drift 0.1m }
{ focusing electric quadrupole, length 0.2m, strength 0.1, aperture 0.05m }
{ drift 0.2m }
{ defocusing electric quadrupole, length 0.2m, strength -0.1, aperture 0.05m }
{ drift 0.1m }
{ magnetic sextupole, length 0.1m, strength 0.01, aperture 0.05m }
{ drift 10m }
{ electric sextupole, length 0.1m, strength -0.01, aperture 0.05m }
{ drift 0.1m }
{ bend magnet, 15m radius, angle 45 degrees, 0.05 aperture }
{ electric condenser, 15m radius, angle 45 degrees, 0.05 aperture }
{ magnetic solenoid, 3A current, 10 turns/meter, 0.01m radius, 0.1m length }
Lattice name:
Syntax help
Lattice parameters
Add new parameter
Add misalignments
Compute lattice with fringe fields
Plot system dependency on parameter
Copyright © 2012–2013 Denis Zyuzin.
About

One cell

rsx
FOR BEAM DYNAMICS

Lattice Designer | Tracking | Run code | Logbook | System status | Configuration | **Sign out**

protons 1000 MeV ▾ Add new...

```
{cell}
mq 0.2 Q_STRENGTH 0.05;
dl 0.1;
magnetic_bend B_RADIUS B_ANGLE 0.05;
dl 0.1;
mq 0.4 -Q_STRENGTH 0.05;
dl 0.1;
magnetic_bend B_RADIUS B_ANGLE 0.05;
dl 0.1;
mq 0.2 Q_STRENGTH 0.05;
```

Lattice name: one cell

Lattice parameters

Name (no spaces allowed)	Value	
Q_STRENGTH	0.3	Remove
B_RADIUS	15	Remove
B_ANGLE	10	Remove

Add new parameter

☐ Add misalignments

☐ Compute lattice with fringe fields

Plot system dependency on parameter

Create new

Load...

Import from MAD-X file...

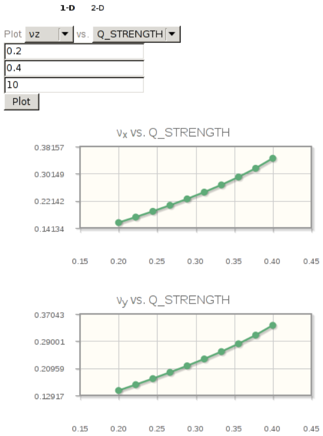
Save

Analyze

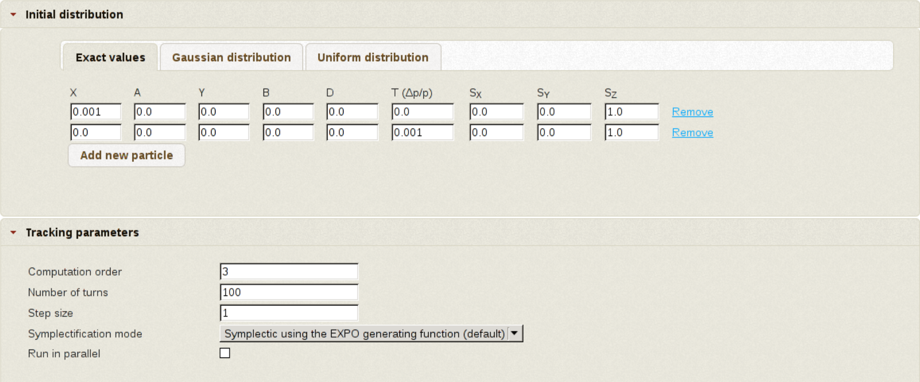
▸ **Settings**

β	0.8750245405539011
γ	2.065780392110563
L_{orb}	6.435987755982989
T_{orb}	2.45343339876033e-8
ν_x	0.2367282701025606
ν_y	0.2267368188567302
ν_z	0
α_1	0.7866773729834815
α_2	0.2521852073908357
η_{xy}	0.08728014481849522
η_{xz}	0.17070894824055105
η_{yz}	0.6555117225306077
ξ_{xy}	0.10899486513593593
ξ_{xz}	0.8621846374679702
ξ_{yz}	0.33110873721406064
$\Delta L/L$	0.7260044573508798

Tunes depending on parameter



Tracking settings



Logbook

Task name	Added	Finished	State
Tracking task	Tue Sep 24 14:02:26 2013	Tue Sep 24 14:02:27 2013	Finished

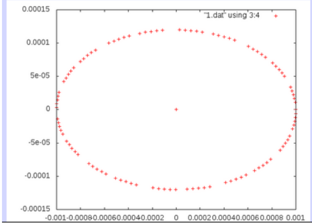
cosyexec task, UUID=**110b79c2-8f11-4fcd-b56c-f8d43d158fa8** on **localhost**
State: **Finished**
Actions: Restart
Start time: **Tue Sep 24 14:02:26 2013**
Finish time: **Tue Sep 24 14:02:27 2013**
Click to show source
Click to show stdout
Click to show stderr
Created files:

- out.ray (view file as plain text or download in gzip archive or plot phase space X-A phase space Y-B phase space T-D S_x rotation S_y rotation S_z rotation)
- out.spin (view file as plain text or download in gzip archive or plot phase space X-A phase space Y-B phase space T-D S_x rotation S_y rotation S_z rotation)

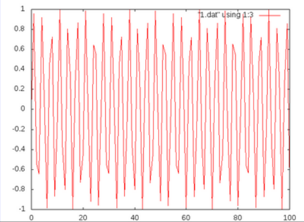
Task name	Added	Finished	State
Compute tunes	Tue Sep 24 12:29:06 2013	Tue Sep 24 12:29:06 2013	Finished
Compute tunes	Tue Sep 24 12:29:06 2013	Tue Sep 24 12:29:07 2013	Finished
Compute tunes	Tue Sep 24 12:29:06 2013	Tue Sep 24 12:29:07 2013	Finished
Calculate length	Tue Sep 24 12:29:05 2013	Tue Sep 24 12:29:06 2013	Finished

Tracking results

Tracking task	Tue Sep 24 14:02:26 2013	Tue Sep 24 14:02:27 2013	Finished



Task name	Added	Finished	State
Tracking task	Tue Sep 24 14:02:26 2013	Tue Sep 24 14:02:27 2013	Finished



System requirements

The server part requires:

- a web-server with FastCGI module, like lighttpd;
- Python 2.6 with SSH, RpyC 3.2, PsycoPG2 modules;
- Django 1.5 or later;
- PostgreSQL database;
- Gnuplot.

The daemon requires only Python 2.x and a Fortran compiler (to build COSY Infinity binary).
Web-side works in all modern browsers with JavaScript support (Mozilla Firefox 21+ or Google Chrome 28+ preffered).

What next

Features under development:

- import lattice from MAD-X;
- spin coherence time calculation, plotting depending on parameters;
- custom analyzing scripts;
- Fourier transform for tune calculation;
- MPI support;
- etc.

Thanks

Thank you for your attention.