

Ramp Optics Development Infrastructure: Problems & Plans

or

Ghosts of RHIC and Other Stories

Al Marusic

2013 RHIC Retreat

OptiCalc Configuration Problem

Strange issues:

- Strengths from a ramp created before RHIC was completed are used by OptiCalc.
- Lengths of gamma jump magnets are not in Holy Lattice files.

How is that possible?

Configuration files in HOLY_LATTICE = /rap/Holy_Lattice directory are binary files in SDS format created by program **dbsf** whose source files are last changed in 1994 and which was last compiled in 2001 (**dbsf** program is part of collection of programs located in /rap/lambda/ directory). **dbsf** generates input files for various beam transport programs from data in **rhic** database (MAD 8, SYNCH, TEAPOT, etc.). That database is supposed to contain the strengths of magnets. In the case of RHIC, we do not store strengths of current ramps in database.

Besides OptiCalc, these files are used by the library for plotting beam elements which is used by RhicOrbitDisplay and other programs (in the case of ATR similar files are used by AtrOrbitDisplay and atremit programs). The contents of these files are also copied into **RHICgddb** database using **sds2db** program. That is part of very complicated procedures for updating **rhic** and **RHICgddb** databases.

How to fix these and other issues?

- Stop making and using these files, instead treat the database tables used to create them in the same way as the tables from RHICgddb database used for the configuration of RampManager are treated, and that is dump them as text files into dbconfig directory associated with the ramp. Then modify OptiCalc to make its own configuration from these files. This will also remove the dependency of operational programs on the files not in /operations directory (i.e. /rap).
- This change allows us to change how OptiCalc works without

affecting old ramps. Confirm that quantities calculated by current and modified OptiCalc for all ramps used during last 3 years are the same.

- Start improving OptiCalc (Guillaume).
- Modify programs using Holy Lattice files to use files from current dbconfig directory or to use data from OptiCalc.
- Procedures for updating **rhic** and **RHICgddb** databases can be simplified and the ramp design can proceed without waiting for those procedures to be performed.
- Currently the programs using Holy Lattice files can not be compiled without special setup. Fix that by importing include files used by OptiCalc into clearcase. After that, those files can be modified to use the same beam element numbering scheme as used by RampManager.
- Document how different beam elements are treated in OptiCalc (Guillaume).
- Document how different beam elements are treated in MAD-X (Guillaume).

- Instead of creating input files for MAD-X before the start of the run, create it for each ramp and store it in the standard place, for example: /operations/app_store/ramps/madx/<ramp name> (maybe do the same with SXF files).
- In these new MAD-X files use the same names for magnets as used by RampManager.
- Allow easy comparison of quantities calculated by OptiCalc and MAD-X.
- Return more data from OptiCalc.

Designing Ramps

E-lens ramps were designed using OpticsDesigner program. Obtained currents looked smooth and tunes straight. However, after making a ramp using those currents, calculated characteristics of the ramp changed.

What stage in ramp making process introduces discrepancies between goal and obtained ramp characteristics (as calculated by OptiCalc)? What causes those discrepancies?

- OptiCalc was not used to design the ramp.
- Does parametrization of strengths used by RampManager introduce errors?
- Something else.

How can we find the answers to these questions?

OptiCalc can not make ramps, so add features to OptiCalc to allow it

to make ramps (i.e. add optimizers to it, ability to store strengths).
Design one of e-lens ramps again, compare it with the ramp designed previously, compare their characteristics, their currents, etc.
Prerequisite for this is making OptiCalc not use Holy Lattice files.

Summary of the Plan:

1. Stop using Holy Lattice files, change OptiCalc to make its own configuration.
2. Start improving OptiCalc, documenting state (Guillaume).
3. Modify OptiCalc to allow it to be used for creating ramps.
Repeat design of e-lens ramp (Vahid).
4. Cleanup database.