A MAD-X MODEL OF THE HIT ACCELERATOR

R. Cee*, M. Galonska, T. Gläßle, T. Haberer, K. Höppner, A. Peters, S. Scheloske

Heidelberger Ionenstrahl-Therapiezentrum, D-69120 Heidelberg, Germany

Abstract

For a medical accelerator facility like the Heidelberg Ion-Beam Therapy Centre (HIT) an online simulation tool with read and write access to the control system and the database is essential for effective beam alignment and beam spot size adjustment at the patient position. Since the commissioning of HIT the simulation programme Mirko from GSI Darmstadt has been in use for the simulation of the beam lines and the synchrotron. While Mirko fully complies with the demands and is still in regular use, the long-term support of the HIT-Mirko derivate cannot be guaranteed. We have therefore started to set up a new simulation environment based on the MAD-X programme from CERN. In a first step we built a MAD-X model of the HIT accelerator using the MAD-X export function of Mirko. The resulting sequences were transformed and extended into executable MAD-X files. The simulation results were validated against Mirko and a good agreement of the calculated beam envelopes could be achieved. Works on the graphical user interface (GUI) for visualisation of and interaction with the beam envelopes and the link to the control system are in progress.

INTRODUCTION

The possibility to predict beam properties in the real machine within a simulation model is of outstanding importance for the design and operation of modern accelerators. Since the commissioning of HIT the programme in use is Mirko, originating from GSI Darmstadt [1]. It is linked to the accelerator control system via a dynamic library providing access to input parameters and beam diagnostics data. Its main application at HIT are correction of the beam position and beam spot size adjustment [2]. Over the years of operation Mirko underwent several improvements and enhancements inspired by the experience of daily operation or driven by changes in the data structure. During all this time HIT did not have direct access to the programme code. In order to counteract the resulting dependency, we started to set up a new simulation environment based on the simulation tool MAD-X from CERN.

MAD-X (**M**ethodical **A**ccelerator **D**esign, hereinafter just "MAD") claims to be the standard for the description of particle accelerators, beam dynamics simulation and beam optics optimisation [3]. The source code – written in C, C++, Fortan77 and Fortran90 – is freely available. MAD is designed as a pure command line interface. This is different from an integrated programme suite like Mirko, which is responsible for both data processing and presentation. At first, this may seem like a burden for users lacking any experience with the tool. But when looking more closely, it is – along with the open source – one of the greatest arguments for choosing the

---

* rainer.cee@med.uni-heidelberg.de

MAD software: the simulation is decoupled from the analysis and presentation of the data. This modularization makes it easy to perform customized post-processing and redirect the data to third party applications as desired, enabling us to develop our own GUI application on top.

Our long term goal is to have a replacement for Mirko in case its support cannot be continued. The basic idea is to set up a complete MAD model of the HIT accelerator, develop a graphical user interface with the Python scripting language and use the same DLL-interface to the control system as Mirko does.

SIMULATION OF THE HIT BEAM LINES

Beam Line Sections

With respect to the simulation, the HIT beam lines shown in Fig. 1 can be divided into the following sections [4]:

- **LEBT**: low energy beam transport from the ion sources to the LINAC (RFQ) input
- **MEBT**: medium energy beam transport from the LINAC (IH-DTL) output to the synchrotron injection
- **Synchrotron**
- **HEBT**: high energy beam transport from the synchrotron extraction to the different targets named horizontal treatment rooms (T1, T2), gantry (T3), experimental area (T4) and beam dump (T5)
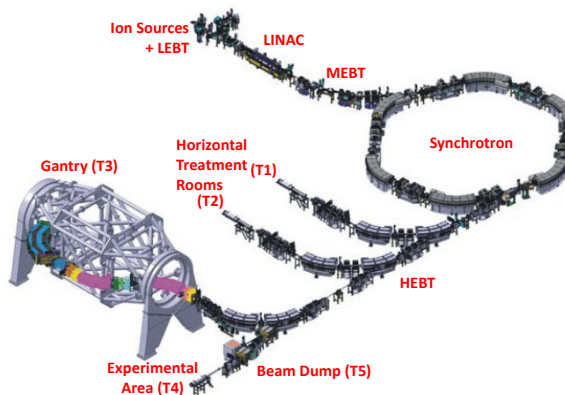

Figure 1: Sections of the HIT accelerator.

Sequence Generation

The core of each MAD model is the beam line sequence. For the translation of the Mirko into the MAD model the built-in export function was used. It generates a MAD sequence of the beam line in the LINE-notation, where only the length of the elements are given and the position is defined by the element order. As we prefer the AT-notation (`at = ...`), with explicit element positioning , the sequences from the export were adapted using a script.

Special attention was paid to the dipoles. We manually brought them into the MAD form DIPEDGE–SBEND–DIPEDGE. We also took care for the right field integral FINT specifying the finite extent of the fringe fields [5]. The global parameter in Mirko (in our case FINT = 0.35) had to be set for each dipole in MAD separately.

In the following envelope plots the Mirko results are regarded as references and are shown in blue lines. The envelopes from the new MAD model are shown in red dots. The upper graphs show the horizontal, the lower ones the vertical beam envelopes. Magnets are displayed as coloured patches (red: dipoles, yellow: quadrupoles, blue: multipoles representing steerers, bumpers and sextupoles). The patches height is not connected to the real aperture of the magnet.

LEBT

The LEBT selects the desired mass-to-charge ratio from the ion source beam by means of a double focusing 90°-dipole and matches the beam to the RFQ input. The pole face curvature of the mentioned dipole is not taken into account in the Mirko model, so we omitted it in MAD as well. A speciality of the LEBT are the two solenoids, which are ignored by the Mirko export function and have to be inserted manually. The first solenoid translates into the following MAD-element:

```
R1MO1: SOLENOID, L=0.235, KS:=KS_R1MO1, at=0.575;
```

where the focusing strength KS in units 1/m is calculated from the magnetic beam rigidity $B\rho$ and the solenoid field B_s from $KS = B_s/B\rho$.
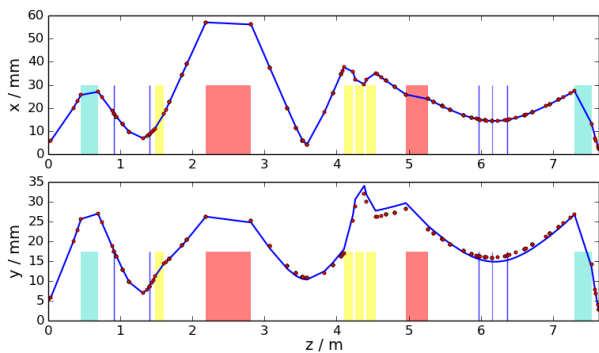

Figure 2: LEBT beam envelopes.

Small discrepancies between MAD and Mirko in the vertical beam envelopes up to ≈1.5 mm can be observed (see Fig. 2). They originate in the 90°-dipole, build up down to the quadrupole triplet but diminish until the end of the section (waist at RFQ input).

MEBT

In the MEBT section the ion beam is stripped and guided to the synchrotron. In the simulation the straggling in the stripper foil (Mirko element "Straggling") has been omitted as there is no equivalent element in MAD. Under this restriction the envelopes show perfect agreement until the debuncher at position 12.631 m (see Fig. 3). The buncher element of Mirko has been translated into MAD the following way:
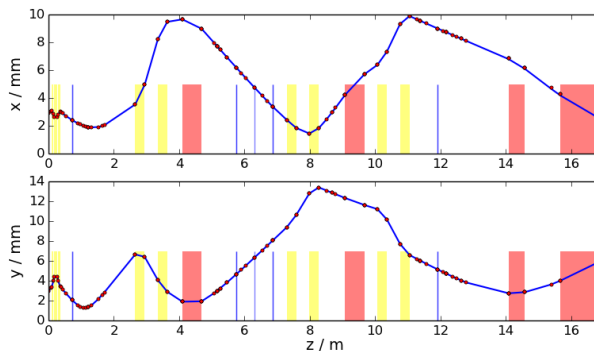

Figure 3: MEBT beam envelopes.

```
M1BB1:RFCAVITY,L=0,VOLT=7550,LAG=0.5,FREQ=217,at=12.63;
```

with the effective buncher voltage VOLT calculated from the focusing strength k as $VOLT = (k \cdot B\rho \cdot (\beta c)^2)/(2\pi\nu)$, where βc represents the ion velocity (here $\beta = 0.122$) and $\nu = 216.816$ MHz being the rf-frequency of the cavity. The phase parameter LAG=0.5 makes the cavity to a debuncher.

On closer inspection the transverse focusing effect of the debuncher is visible in the Mirko simulation, whereas this effect seems to be neglected by MAD. Admittedly, this difference does not produce a considerable deviation.

Synchrotron

The synchrotron accumulates the beam during a multiturn injection and accelerates the ions to the requested extraction energy [6]. In the final stage of each cycle the beam is slowly extracted by means of a KO-extraction. Two different beam optics are applied for the injection and extraction phase, the one shown in Fig. 4 being the injection mode. Both models show very good agreement in both planes.
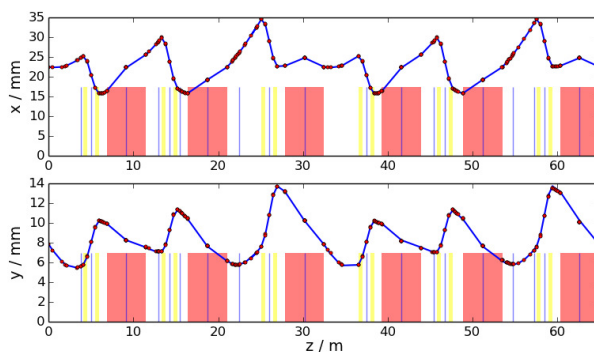

Figure 4: Synchrotron beam envelopes. The correction coils of the 60°-dipoles are modelled as thin steerers in the magnet centres.

HEBT and Gantry

The HEBT guides the beam to the three treatment rooms and the experimental area. These models are most frequently used in machine operation as they are important for beam alignment and beam size adjustment at the patient treatment position. The beam lines to the fixed beam rooms are not challenging with respect to the simulation. However, the beam line to the gantry might include coordinate rotations.

MOPME010

398

05 Beam Dynamics and Electromagnetic Fields

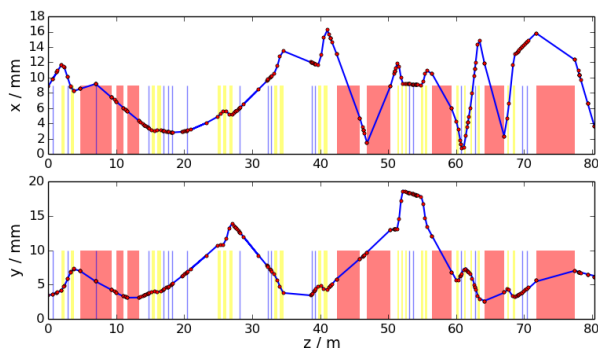**D06 Code Developments and Simulation Techniques**

Figure 5: HEBT beam envelopes up to T3 for a gantry angle of 90° (accelerator plane).

The envelopes with the gantry in 90° position, corresponding to the accelerator plane, is shown in Fig. 5.

MADGUI

The possibility to have a GUI programme for the specific needs at HIT has played an important role from the beginning. Interoperability with the online control system is a high priority feature at HIT. Furthermore, we want to be able to quickly adapt to future demands. These considerations where taken into account when starting the development of a new GUI application, called MadGUI [7].

The programme is currently able to load MAD files, forward them to the MAD library, fetch back the results and conveniently display them inline in an application provided window, looking to the user much similar to the familiar Mirko programme. Every application window has a dedicated shell tab where arbitrary python commands can be entered and the MAD interpreter can be controlled directly. This makes the programme useful even if the built-in features turn out to be insufficient for a specific use-case. Furthermore, there can be an arbitrary number of tabs with different views for data analysis of the simulated model, or for presenting any other related information. The display window
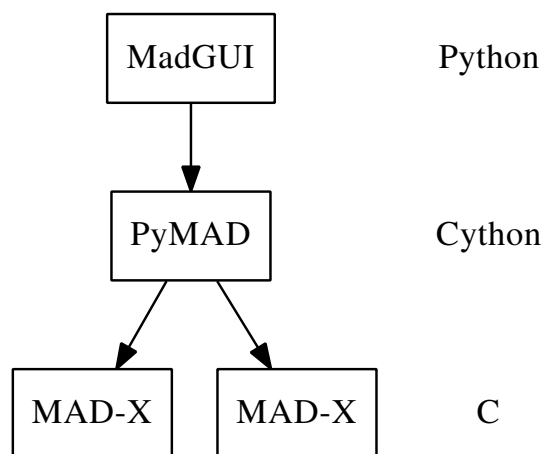

Figure 6: Overview of the MadGUI key components. The programme is designed to handle several MAD instances simultaneously.

is interactive allowing the user to match the beam envelope via mouse clicks or view the parameters of specific beam line elements.

Essentially, the back-bone of MadGUI is a plugin system written in python. This makes it easy to dynamically add specialized components, like the online control interface in our case. There are core modules using the well-established matplotlib library for data display on a wxPython based GUI. As illustrated in Fig. 6 the communication to the MAD back-end is moderated by a python binding called PyMAD [8], in which we invested considerable development effort as well.

OUTLOOK

We made the software free and entirely open-source in the hope that the work done at HIT may be useful to other parties and in the long term the project may profit from patches and the influence of the scientific community. The programme is designed to be cross-platform, working on Windows and GNU/Linux systems alike.

The next step will be the integration of read and write capability of element parameters into MadGUI. Once this is running the implementation of beam alignment procedures under usage of existing MAD functionality will be of highest priority.

ACKNOWLEDGEMENTS

REFERENCES

[1] B. Franczak, ion optics simulation programme MIRKO, GSI Darmstadt.

[2] D. Ondreka, U. Weinrich, "The Heidelberg Ion Therapy (HIT) Accelerator Coming into Operation", Proc. EPAC 2008, Genoa, Italy, 2008, pp. 979–981.

[3] MAD-X, Methodical Accelerator Design, CERN-BE/ABP Accelerator Beam Physics Group, http://mad.web.cern.ch/mad/.

[4] H. Eickhoff et al., "HICAT - The German Hospital-Based Light Ion Cancer Therapy Project", Proc. PAC 2003, Portland, U.S.A., 2003, pp. 694–698.

[5] Karl L. Brown. A First-and Second-Order Matrix Theory for the Design of Beam Transport Systems and Charged Particle Spectrometers. SLAC 75, Revision 3, SLAC, 1972.

[6] A. Dolinskii, H. Eickhoff, B. Franczak "The Synchrotron of the Dedicated Ion Beam Facility for Cancer Therapy, Proposed for the Clinic in Heidelberg", Proc. EPAC 2000, Vienna, Austria, 2000, pp. 2509–2511.

[7] https://github.com/coldfix/madgui

[8] K. Fuchsberger, Y. Inntjore Levinsen, "PyMad – Integration of MadX in Python", Proc. IPAC 2011, San Sebastián, Spain, 2011, pp. 2289–2291.