

RAY-UI: A Powerful and Extensible User Interface for RAY

P. Baumgärtel^{1, a)}, M. Witt², J. Baensch³, M. Fabarius³, A. Erko¹,
F. Schäfers¹ and H. Schirmacher³

¹*Institute for Nanometre Optics and Technology*

²*Department Operation Accelerator BESSY II*
Helmholtz Zentrum Berlin für Materialien und Energie
Albert-Einstein-Str. 15, D-12489 Berlin,

³*Fachbereich VI – Informatik und Medien*
Beuth Hochschule für Technik Berlin,
Luxemburger Str. 10, D-13353 Berlin

^{a)}Corresponding author: peter.baumgaertel@helmholtz-berlin.de

Abstract. The RAY-UI project started as a proof-of-concept for an interactive and graphical user interface (UI) for the well-known ray tracing software RAY [1]. In the meantime, it has evolved into a powerful enhanced version of RAY that will serve as the platform for future development and improvement of associated tools.

The software as of today supports nearly all sophisticated simulation features of RAY. Furthermore, it delivers very significant usability and work efficiency improvements. Beamline elements can be quickly added or removed in the interactive sequence view. Parameters of any selected element can be accessed directly and in arbitrary order. With a single click, parameter changes can be tested and new simulation results can be obtained. All analysis results can be explored interactively right after ray tracing by means of powerful integrated image viewing and graphing tools. Unlimited image planes can be positioned anywhere in the beamline, and bundles of image planes can be created for moving the plane along the beam to identify the focus position with live updates of the simulated results.

In addition to showing the features and workflow of RAY-UI, we will give an overview of the underlying software architecture as well as examples for use and an outlook for future developments.

INTRODUCTION - MOTIVATION

RAY is a well-known ray tracing program and design tool for synchrotron radiation beamlines and for other optical systems [1]. In the beginning of the development of RAY in the year 1984 its user interface concept was based on interactive sequential input of the model parameters via the command line. The graphical visualization of the results after tracing was performed by launching a separate bitmap viewing application. Both concepts were state of the art in the early years of developing RAY, but nowadays users are expecting software tools with intuitive and ergonomic graphical user interfaces and fully integrated, end-to-end workflow. These requirements were the primary reasons for starting the RAY-UI project. The main additional constraint was not to compromise any of the unique features of the existing RAY software, including:

- Implementation of realistic sources, in particular SR-sources (Dipole, Undulator...)
- Covering a large set of synchrotron radiation optical elements
- Broad spectral range of optical constants included
- Clever parameter suggestions
- Tracking path lengths for simulation of interference and wavefront effects

USABILITY AND WORKFLOW IMPROVEMENTS

Based on the optical designer's workflow we have designed three major views for interacting with a virtual beamline, see Fig.1. The designer obtains an overview of the sequence of beamline elements in the so-called *Sequence View*. Here the optical system can be set up and modified interactively. The model parameters for each individual element are available in the *Parameter View*. After tracing rays through the system, the results are provided in the *Analysis View*. For certain operations, graphical guides help operating the application, especially for novel users.

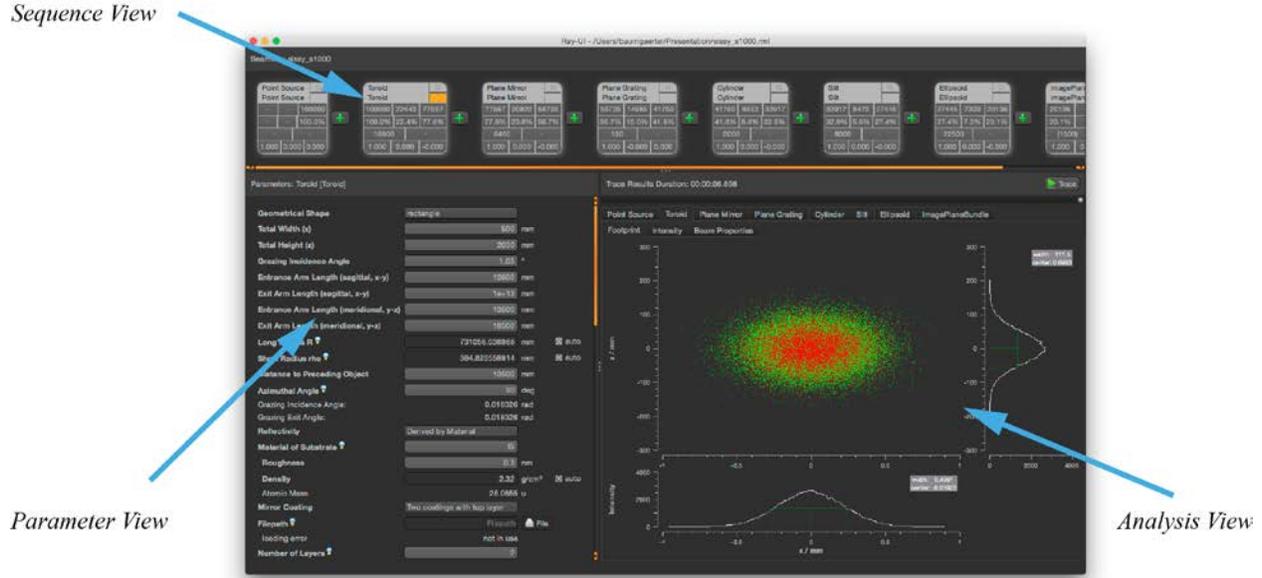


FIGURE 1. Main window of RAY-UI after tracing process

Setting Up the Line Using the Sequence View

The *Sequence View* displays the whole optical system from left to right, including source and optical elements. The sequence of the elements follows the propagation of light in the optical systems. Any type of element can be added at any position in the beamline by clicking one of the “+” buttons and choosing the element from a list. Removing an element is just as easy by clicking on the “x” in the upper right corner of the element (see Fig. 2).

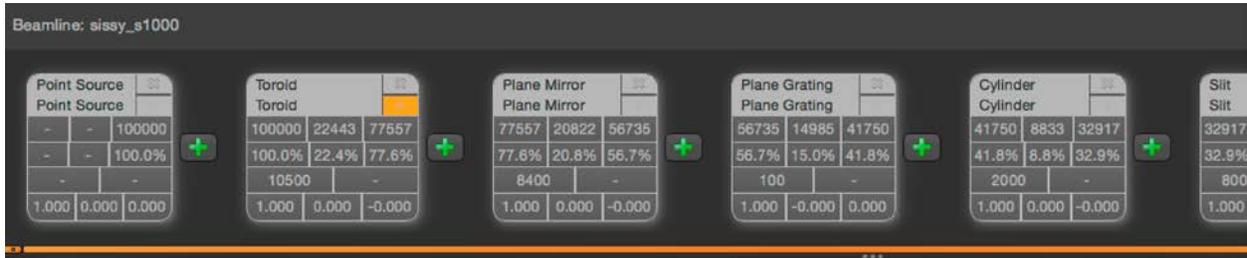


FIGURE 2. *Sequence View*. In this version of RAY-UI basic ray tracing results are tabularized for each element in this view: the first line contains the absolute number of incoming, absorbed and outgoing (reflected or transmitted depending on the element) rays. The second contains the same but in percent relative to the source. The number in line three is the distance to the preceding element in mm. The averaged Stokes vector ($S1/S0$, $S2/S0$ and $S3/S0$) is shown in the last line.

Efficient Interaction with Smart Parameters

The model parameters of the currently activated element (marked in orange in the *Sequence View*) are listed in the *Parameter View*. They are accessible in random order, and some of them are functionally interconnected. Changing one such parameter will cause an instant update of all dependent parameters. Automatic value suggestion is provided for some parameters by simply checking the respective “auto” check box. Additional or alternative parameter sets can be activated by comfortable pull-down menus that will cause instant update of the relevant parts of the user interface. In this first version of RAY-UI we stay very close to the original naming and order of RAY parameters in order to facilitate the migration from RAY to RAY-UI.

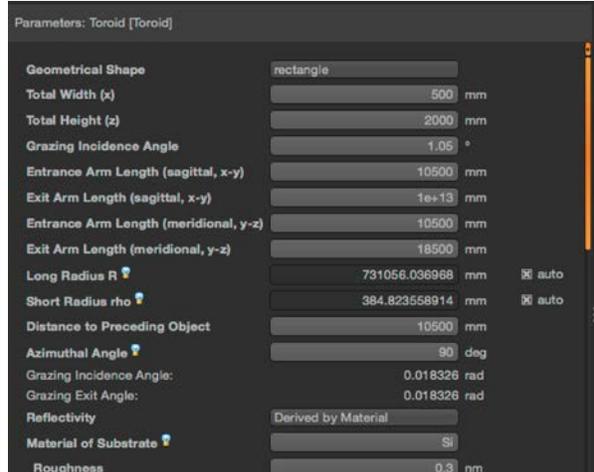


FIGURE 3. *Parameter View*, lamp icons indicate additional information about the parameter, shown in a pop-up balloon

Ray Tracing and Interactive Exploration of Results

Clicking on the “Trace” button will start the ray tracing process and display a progress bar. In contrast to the original RAY software the tracing process itself does not require any human intervention and thus can be performed as a true background task. Tracing is done element by element, and the raw data are stored within main memory for efficient further processing.

RAY-UI allows instant visualization and interactive exploration of the rays’ intersections with each element or image plane in the beamline. The intersection points on each surface are displayed in 2D as colored footprint plots or colored intensity (2D histogram) plots. Energy, path length, and ray directions are presented as distribution histograms (including width and peak position). The footprint plot uses colors to distinguish outgoing rays (reflected or transmitted, depending on the optical element) from absorbed and “wasted” rays (green, red, and grey, respectively). The term “wasted” refers to rays that fail to intersect the surface inside of the given element’s area. This new feature helps the designer to control the illumination of the optical element. Pan and zoom operations are available independently for each of the two-dimensional plots. Views can be scaled either proportionally or using a “fit to window” mode via the context menu.

Instant Virtual Image Planes

RAY-UI allows the designer to insert “Image Planes” as virtual screens at the end of the beamline or within any two elements. These screens visualize the ray distribution at any point in the beamline and have no impact on the actual optical elements. Computation and visualization of the virtual screen is immediate, without need for an additional ray-tracing step.

RAY-UI also introduces a “movable” screen called “Image Plane Bundle”, which can be placed anywhere into the beamline in the same way. The bundle allows moving the image plane position using an interactive slider with close to real-time update of respective beam. This extension can be used for visualizing important beam properties such as the focus (see Fig. 4).

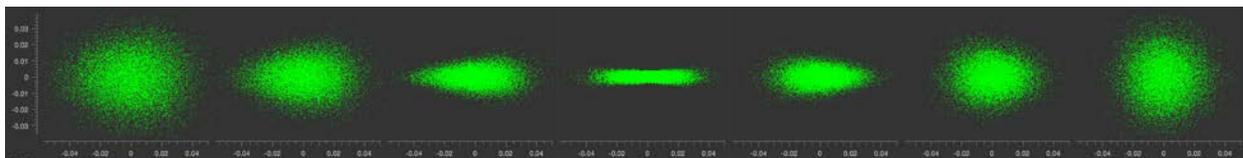


FIGURE 4. Snapshots of a series of screens derived by the “Image Plane Bundle”, the screen was moved through the focus in a range of +/- 10 mm in 7 steps.

Import, Export and Print Capabilities

Ray-UI allows easily saving and opening of beamlines including their complete parameter sets as human-readable XML (Extensible Markup Language) markup files. Furthermore, all or selected computational results can be exported as CSV (Character Separated Values) files. A matrix of all optical elements and all available calculation results allows the user to customize the output by simply clicking on the desired items.

Plots can be saved to image files in a multitude of image formats. Comprehensive reports combine graphical and numerical information and can be printed or exported in PDF format. The user can easily save and sort different exported data by automatically suggested filenames.

Auto Tracing Parameter Ranges

A further new feature called “Auto Trace” allows the user to scan over sets or ranges of parameter values. The context menu of a parameter (right mouse click on a parameter in the *Parameter-View*) allows adding this parameter to the auto tracer. In this tool the user can configure the parameter values or ranges over which the tool will automatically scan while tracing the beamline.

SOFTWARE ARCHITECTURE AND PLATFORM

RAY-UI is developed in C++ using the Qt (<http://www.qt.io/>) and Qwt (<http://qwt.sourceforge.net/index.html>) framework. To us this choice seems to be the best-balanced platform on which physicists on one side and software and UI engineers on the other side can collaborate efficiently and in the long term.

A lot of care was given to the integration of RAY with RAY-UI. The current RAY Fortran code can be plugged “as is” into the RAY-UI platform. As a result, RAY-UI can easily benefit from any updates in the RAY code.

CONCLUSION AND FUTURE PLANS

RAY-UI, with its intuitive and efficient user interface and workflow, is the logical extension of the popular RAY software for the years to come. Already today RAY-UI can be used as fully functional replacement of RAY. Many more enhancements are planned, such as support for new sources and optical elements, phase and interference effects, further workflow improvements, and research on speeding up the ray tracing core on modern graphics hardware.

WHERE TO GET

The Installer can be downloaded from the application’s home page located at the Helmholtz-Zentrum Berlin: <http://hz-b.de/ray>. We plan to provide two kinds of installers, one will contain the current release - a stable version, and the other one will contain a developer’s version with newest features for testing.

ACKNOWLEDGMENTS

We would like to thank all current and future users for their help with testing and improving RAY-UI. Please feel free to contact us with your bug reports, concerns or ideas.

REFERENCES

- [1] F. Schäfers, “The BESSY Raytrace Program RAY”, in *Modern Developments in X-Ray and Neutron Optics, Springer Series in Modern Optical Sciences*, edited by A. Erko, M. Idir, Th. Krist, A.G. Michette, (Springer-Verlag, Berlin Heidelberg, 2008), Vol. 137, p. 9