

THE PARTICLE BEAM OPTICS INTERACTIVE COMPUTER LABORATORY FOR PERSONAL COMPUTERS AND WORKSTATIONS

George H. Gillespie, Barrey W. Hill, Hendy Martono, John M. Moore,
Nathan A. Brown, Michael C. Lampel and R. Chris Babcock
G. H. Gillespie Associates, Inc., P.O. Box 2961, Del Mar, CA 92014, U.S.A

Abstract

The Particle Beam Optics Interactive Computer Laboratory (PBO Lab) is a new software concept to aid both students and professionals in modeling charged particle beams and particle beam optical systems. The PBO Lab has been designed to run on several computer platforms and includes four key elements: (1) a graphic user interface shell, (2) a graphic beamline construction kit for users to interactively and visually construct optical beam lines, (3) a knowledge database on electric and magnetic optics elements, including interactive tutorials on the physics of charged particle optics and on the technology used in particle optics hardware, and (4) a set of charged particle optics computational engines that computes transport matrices, beam envelopes and trajectories, fits parameters to optical constraints, and carries out similar calculations for the graphically-defined beam lines. The primary computational engines in the first generation PBO Lab are the third-order TRANSPORT code, the multiple ray tracing program TURTLE, and a new first-order matrix code that includes an envelope space charge model with support for calculating single trajectories in the presence of the beam space charge. Progress on the PBO Lab development is described and illustrations from the Windows 95 implementation are presented.

1. INTRODUCTION

The increasing variety of accelerator applications in research and industry places a growing demand upon scientists and engineers involved in developing new accelerator and beamline designs. Particle optics simulation codes play a key role in the design process and enhanced software tools offer the promise of improved productivity for beamline designers. Another benefit of improved software tools is a significant reduction in the time required to train new researchers in the use of optics simulation programs. This paper summarizes progress on one such new tool: The Particle Beam Optics Interactive Computer Laboratory (PBO Lab). An object oriented programming (OOP) approach has been adopted for the

PBO Lab and the architecture of the software has been developed specifically to run on several different operating systems. An overview of each of the primary components of the PBO Lab has been presented elsewhere [1]. Here we describe the first two key elements of the PBO Lab listed in the abstract: the graphic user interface shell and the beamline construction kit.

One emphasis in the PBO Lab is on assisting users in the set up and running of the optics programs without requiring any knowledge of the format, syntax, or similar requirements of the input. Beamlines and accelerator systems are graphically constructed on the computer screen using drag and drop icons. Default parameters are incorporated for all required inputs so that both the topology of the beamline and a complete set of input data are defined automatically during the graphical construction. Setting up a particular design is reduced to editing the values of parameters, which are displayed in windows together with the parameter descriptions. Different parameter set options are available for defining optical elements, and a variety of units options may be used. Expert system type rules provide guidance for editing input parameters, and additional displays, such as effective focal lengths and phase space plots, provide users with further useful feedback on their input. The graphic user interface shell and the beamline construction kit focus on these objectives.

2. GRAPHIC USER INTERFACE (GUI) SHELL

The conceptual foundation for the PBO Lab GUI shell is derived from a unique graphic user interface designed specifically for codes used in the accelerator community, known as the Shell for Particle Accelerator Related Codes (S.P.A.R.C.) [2]. A new multi-platform (MP) version of the S.P.A.R.C. environment has been developed to support a number of different operating systems [1]. S.P.A.R.C. MP is written in C++ and has been constructed using a modular approach that provides a good framework for implementing a cross platform version of the GUI. Figure 1 illustrates the PBO Lab GUI running on the Windows 95 operating system.

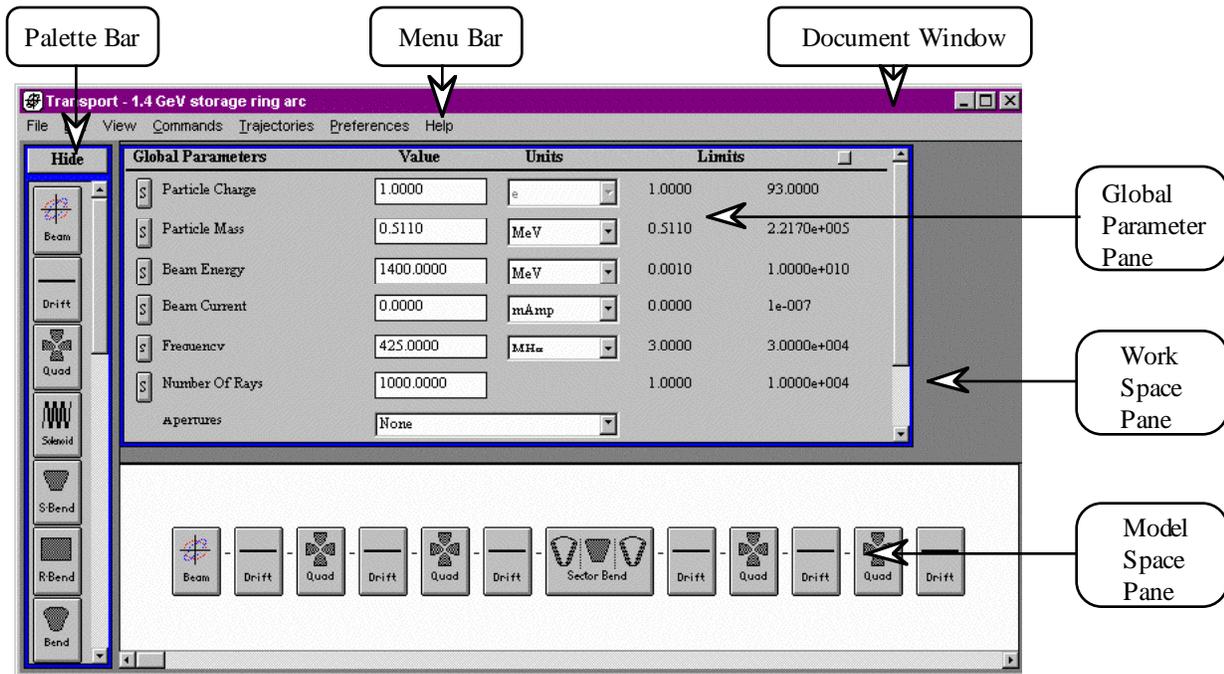


Figure 1. Example of the PBO Lab graphic user interface for TRANSPORT using S.P.A.R.C. MP.

Parameter values are edited using Data Tables in the Global Parameter Pane or in Piece Windows for individual beamline components. A Piece Window is accessed by “double clicking” on the icon of the desired beamline

component. Figure 2 illustrates a Piece Window for the Quad element of TRANSPORT [3] or TURTLE [4]. The PBO Lab Piece Windows have several features that assist users in editing input data.

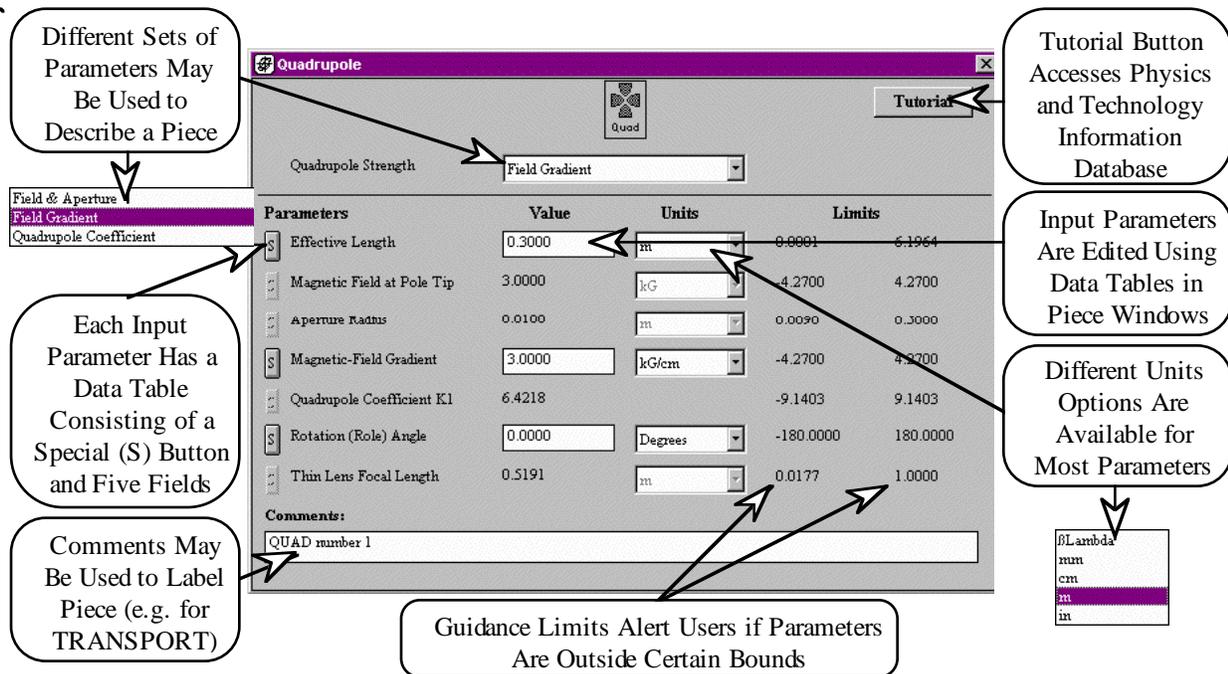


Figure 2. Example of PBO Lab Piece Window for editing parameters of a magnetic quadrupole element.

3. BEAMLINE CONSTRUCTION KIT

The construction of a beamline using S.P.A.R.C. MP follows the basic approach used in the original S.P.A.R.C. environment [1]. A beamline is built by selecting and dragging (with the mouse) the icons from the Palette Bar for individual transport elements and dropping them onto the Model Space Pane. Graphical representations of components (Piece icons) are then displayed on this pane. Figure 1 illustrates a Model Space beamline composed in this way that contains drifts, quadrupoles and a bending magnet. Transport elements (Pieces) can be snapped to either end of the beamline and may also be inserted into the middle of the beamline, by selecting a drop location that is near one of the connection lines between elements on the beamline.

Individual Pieces or groups of Pieces on the beamline may be selected for use in other beamline

construction tasks. Once a selection is made, it may then be Copied, Cut, Deleted, or dragged up to the Work Space Pane for future use. Pieces from the Work Space may be inserted in, or dropped onto the ends of, the beamline on the Model Space. Selected Pieces may also be defined as "Sublines" to be used in constructing beamlines composed of repetitive elements (e.g. a lattice). A number of operations on Sublines are being developed, such as Expand, Invert and Rotate operations. A powerful object model has been developed [5], which very efficiently describes either hierarchical (Subline), flat (single Piece), or mixed beamline representations, that forms the basis for this graphic functionality. Figure 3 illustrates an example of Subline use, where the arc shown in Figure 1 has been defined as a Subline and then replicated 5 times. Sublines may contain additional Sublines, as well as individual Pieces.

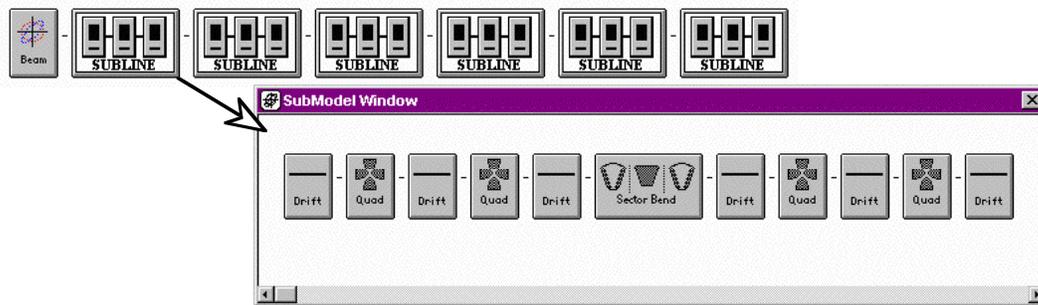


Figure 3. Using Sublines to construct a beamline composed of a repetitive group of elements.

4. SUMMARY

The Particle Beam Optics Interactive Computer Laboratory promises improved productivity for scientists and engineers involved in the analysis or design of accelerators and a significant reduction in the time required to train new researchers in the use of optics programs. Substantial progress has been made in developing the PBO Lab into a useful tool, including the integration of TRANSPORT into the S.P.A.R.C. MP environment running in the Windows and Solaris operating systems.

ACKNOWLEDGMENTS

The authors are indebted to Dave Carey for assistance in defining the requirements for the PBO Lab needed to support the TRANSPORT and TURTLE codes. We also thank Karl Brown for useful comments on early versions of the PBO Lab and TRANSPORT GUI. The assistance of James Gillespie in the design of the software architecture for the PBO Lab is gratefully acknowledged. Portions of this work have been supported by the U. S. Department of Energy under SBIR grant number DE-FG03-94ER81767. Windows is a trademark of Microsoft and Solaris is a trademark of Sun Microsystems.

REFERENCES

- [1] G. H. Gillespie, B. W. Hill, N. A. Brown, H. Martono and D. C. Carey, "The Particle Beam Optics Interactive Computer Laboratory," AIP Conference Proceedings 391, 264-269 (1996).
- [2] G. H. Gillespie, "The Shell for Particle Accelerated Related Codes (SPARC) - A Unique Graphical User Interface," AIP Conference Proceedings 297, 576-583 (1993).
- [3] D. C. Carey, K. L. Brown and F. Rothacker, "Third-Order TRANSPORT - A Computer Program for Designing Charged Particle Beam Transport Systems," Stanford Linear Accelerator Center Report No. SLAC-R-95-462, 295 pages (1995).
- [4] D. C. Carey, "TURTLE (Trace Unlimited Rays Through Lumped Elements) A Computer Program for Simulating Charged Particle Beam Transport Systems," Fermi National Accelerator Laboratory Report No. NAL-64, 45 pages (1978).
- [5] B. W. Hill, H. Martono and J. S. Gillespie, "A Object Model for Beamline Descriptions," AIP Conference Proceedings 391, 361-365 (1996).