ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S16MS03

The LEP Model Interface for MAD

F. Christoph Iselin
CERN, SL Div.
CH-1211 Geneva 23
Switzerland

Abstract

During machine studies and trouble-shooting in the LEP machine various optical parameters must be computed, which can be found quickly using the MAD program. However, the LEP operators are not all well acquainted with MAD. In order to ease their task, a simple interface called the LEP model has been written to run on the Apollo workstations of the LEP control system. It prepares jobs for MAD, sends them to a DN 10000 node for execution, and optionally plots the results.

The desired machine positions and optical parameters vary between LEP runs. The LEP model contains a powerful selection algorithm which permits easy reference to any combination of positions and optical parameters in the machine. Elements can be chosen by name, by sequence number, or by element class. The choice of optical functions includes closed orbit, Twiss parameters, betatron phases, chromatic functions, element excitations, and many more. Recently matching features have been added.

Communication with the control system and with MAD uses self-describing tables, i. e. tables whose columns are labelled with their name and a format code. Experience with this LEP model interface is reported.

1 Introduction

This section describes those aspects of the LEP control system and of MAD which are relevant to the LEP model program. The second section outlines features of the LEP model program. The third section discusses implementation, and the last two sections present future plans and experiences with the program.

1.1 LEP as seen from LEP Model

The LEP control system [1] is based on a network of Apollo workstations connected in a token ring network. The workstations are running under UNIX. They talk to the LEP machine over various links and microprocessors. For time-intensive tasks the network contains an Apollo DN 10000 computer, whose speed is about a factor 1/2 of the IBM 3090.

The descriptions of the LEP machine and of its possible optical configurations reside in an Oracle database. From the database a structural description of LEP is available which is formatted in MAD input language.

For equipment control the access to the Oracle database is too slow. A set of files, known as the "reference data set", is thus extracted and stored in a file server. Most of these files are self-describing tables, known as TFS tables (Table File System [5]). Each table has an arbitrary number of descriptors, and each column is labelled with its name and format code.

The status of equipment, e. g. the magnet excitations, or the RF cavity settings, can be acquired via specialized programs and is usually stored in TFS format. TFS tables can also be sent to LEP to modify the settings of equipment.

1.2 MAD seen from LEP Model

The MAD program [2, 3] has been used extensively for the design of LEP. It is based on a "standard language" [4], used to describe the machine structure, and to request various computations on this structure. The language is designed to make communication with a human user easy. For communication with other programs MAD also understands TFS format.

In the framework of the LEP Model Program MAD serves the following purposes:

- Compute the closed orbit,
- Compute optical functions over parts of the machine,
- Match optical functions to specific conditions,
- Calculate global parameters of LEP,
- Change machine parameters to study their effect.

2 The LEP Model Program

2.1 Tasks

Based on the above, the LEP Model Program must

- Use the reference data set to build menus of available optical configurations and to present them to the user for choice. In this way the program needs no changes if new configurations are installed. The proposed default is taken from a file known as the LEP Run-Table.

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S16MS03

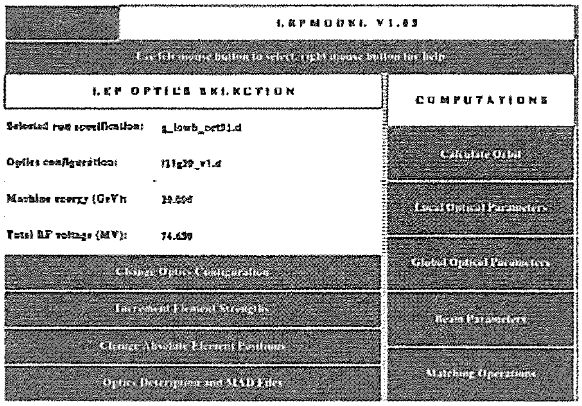

Figure 1: Opening Screen for LEP Model Program

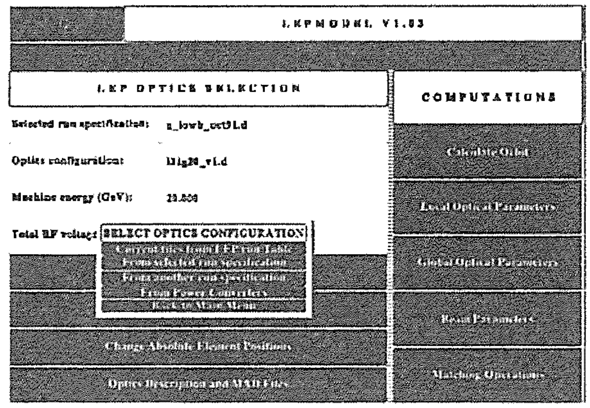

Figure 2: Menu to Change Optical Configuration

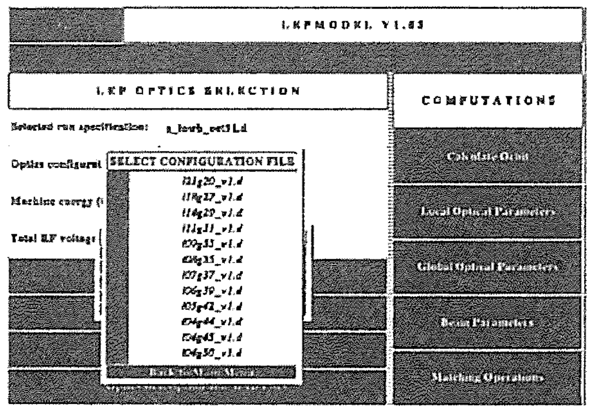

Figure 3: Menu of Available Optical Configurations

- Deliver the following files to MAD:
  1. The definition of elements (magnets, cavities, etc.) in the machine, and the sequence of their occurrences. This file is built in the Oracle database and formatted in the standard MAD input language. It only changes when the LEP machine is modified physically.
  2. The current optical configuration (magnet excitations). This file is a TFS table selected from the reference data set, or built by reading the actual power converters by launching a specialized program.
  3. The "imperfection file". This file contains known imperfections of the machine, like measured multipole components in the machine dipoles. It is updated manually when such imperfections become known.
  4. The "trim file" introduces factors to compensate observed errors of the model. The corrections are constructed by trial and error, and do not necessarily represent actual imperfections of LEP.
- Allow interactive increments of element excitations.
- Allow interactive assignment of element displacements.
- Set up a MAD command file to read the above files, and to perform the desired computations.
- Launch MAD on the DN 10000 server.
- Present a table of results on the display.
- Optionally plot the results on the display.

In the standard version of the LEP Model Program all operations must be done in such a way as to avoid MAD to fail during computation.

2.2 Examples of Menu Selections

The opening screen of the LEP Model Program is shown in Figure 1. Selection of the "Change Optics Configuration" icon pops the menu shown in Figure 2. It presents options for selecting configuration files. Selection of "From current run specification" then presents the menu of Figure 3, containing all states for the current run. To compute the optical parameters in selected positions the user first selects "Local Optical Parameters" from the main menu and gets the menu shown in Figure 4 After selection of "Select Elements" the program pops the menu shown in Figure 5 which allows to select the positions where to calculate the parameters. Selection of "Select Parameters" then presents the menu of Figure 6 offering the available optical functions. While it is computing, the LEP Model Program displays a screen like in Figure 7. The results may be plotted as shown in Figure 8.

3 Implementation

3.1 Choice of User Interface Package

Domain/Dialogue was chosen for the following reasons:

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S16MS03

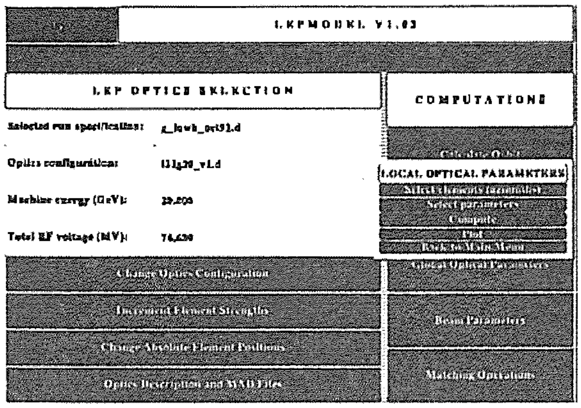

Figure 4: Menu for Local Optical Parameters

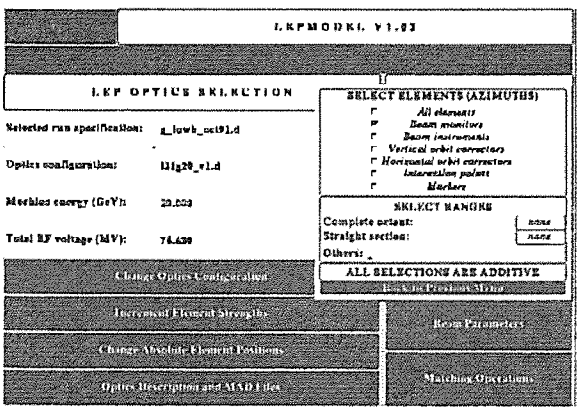

Figure 5: Menu to Select Element Positions in the Machine

- Dialogue allows to define and modify the user interface independently of the application program. The user interface can be built and tried before writing a lot of code.
- Dialogue is quite powerful, and easy to use and maintain.
- Dialogue has been used extensively at CERN.
- Since most of the workstations are Apollos, there is no problem in using a non-portable package.

Dialogue can be used with FORTRAN, Pascal and C programs. FORTRAN requires use of non-standard types like **INTEGER*2**, and of contorted **EQUIVALENCE** constructs to simulate Pascal or C structures. Pascal is little used at CERN. C has been chosen chiefly because it is the main programming language used in the LEP control system.

3.2 Running MAD on a Remote CPU

The LEP Model Program requires a graphics display, thus it must run on the user's workstation. For speed reasons MAD must run on a more powerful computer, like the DN 10000.

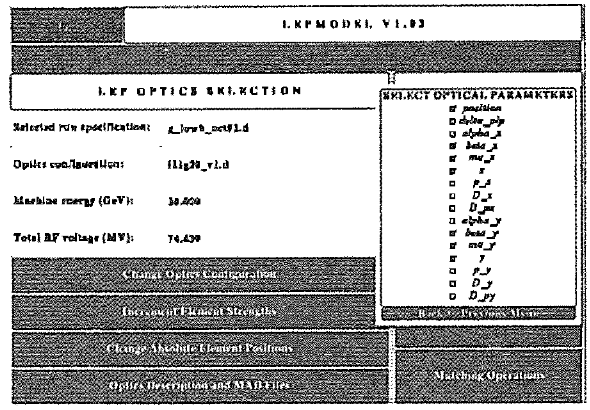

Figure 6: Menu to Select Output Parameters

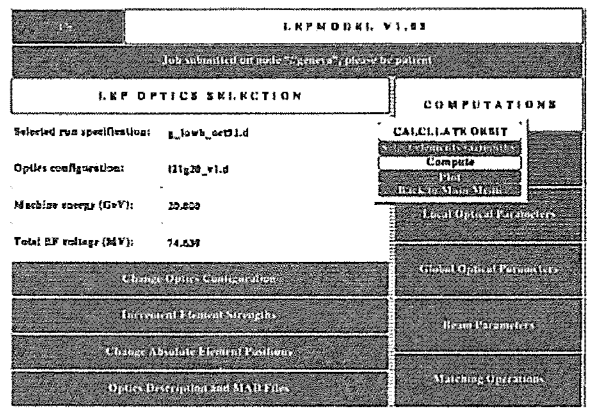

Figure 7: The LEP Model Program is Computing...

Unfortunately one cannot use the UNIX standard system call (nor fork + execl) to spawn another process on th DN 10000. It was found that under the current release of the Apollo UNIX system (SR 10) these calls may crash the operating system. MAD is therefore launched by the Aegis command crp (create process). This command can use the current login name and password of the user, and can be called by the program in a transparent way.

3.3 Data Transfer

The UNIX file system makes it easy to access files from remote nodes. All data transfer uses ASCII files residing in the user's current directory. MAD input files are in ASCII TFS format where feasible, or in MAD input language. MAD output files are written in MAD language, as TFS files, or as listings. This solution allows to develop and test the LEP Model Program independently of MAD by looking at those ASCII files.

3.4 Plotting

Most results come out of MAD as a TFS table. On request the LEP Model Program invokes the LEP dataviewer in a

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S16MS03

Figure 8: Example of Closed Orbit Plot (one quadrupole displaced)

new window, to plot the newly computed data. The user is free to use dataviewer tools (graph selection, zoom ...) to study the shape of the curves, and a hard-copy of the screen can be made on a Tektronix printer.

4 Future Plans

The LEP Model Program will mainly have to be expanded in the area of interaction with the control system. Operation requires more choices for matching machine parameters, and for introducing the new excitations into the machine. For safety the changes will be introduced by hand, and possibly in small increments; thus effectively creating a "knob" which can be turned to apply a correction to any percentage varying from zero to full value.

In future the program should also generate complete sets of files which can be used to set up a new configuration. Possible options are the following:

- Match excitations to achieve the desired behaviour of the machine, and return the excitation file.
- Compute a table of optical parameters for the machine.
- Launch a program to split the table of optical parameters and to feed the parts into various components of the control system, e. g. the closed orbit correction program.
- Launch a program to set the magnet excitations.
- Launch a program to compute the effect of quadrupole and sextupole strengths onto machine tunes and chromaticities.

5 Conclusions

The LEP Model Program has been found to be a very useful tool for machine setup, as well as for testing new optical configurations during machine development. During operation it has been used successfully for calibration of equipment, to test hypotheses about misalignment and mispowering; this has contributed to the comprehension of various effects in the machine.

The accelerator physicists doing machine development usually have a better knowledge of the MAD program, and they wish to have more freedom in using the LEP Model. A special version of the LEP Model is provided which allows to use private files and/or to edit the files provided by the system. However, this version offers little protection against use of wrong data.

The choice of Domain/Dialogue makes this program modular and expandable. Since the system is event-driven, changes are very simple. To add a new feature it is often sufficient to add a few lines to the interface file, to write and compile a new C routine, and to relink the program without touching any existing C code. Even major rearrangements of the menus are feasible by editing the interface file and by relinking without recompilation.

References

[1] P. G. Innocenti, "The LEP Control System". *International Conference on Accelerator and Large Experimental Physics Control Systems*, Vancouver, Canada, October 30–November 3, 1989.

[2] H. Grote and F. C. Iselin, *The MAD Program, Version 8.4, User's Manual.* CERN/SL/90-13 (AP), Rev. 2, August 1991.

[3] H. Grote and F. C. Iselin, "Recent Developments in MAD Version 8". *International Conference on Accelerator and Large Experimental Physics Control Systems*, Vancouver, Canada, October 30–November 3, 1989.

[4] D. C. Carey and F. C. Iselin, "A Standard Input Language for Particle Beam and Accelerator Computer Programs". *1984 Summer Study on the Design and Utilization of the Superconducting Super Collider*, Snowmass, Colorado, June 23–July 13, 1984.

[5] Ph. Hofmann, *Table File System, Reference Guide.* CERN/LEP LAW Note No. 9 (unpublished).