

**GENI: A GRAPHICAL ENVIRONMENT FOR MODEL-BASED CONTROL\***

STEPHEN KLEBAN and MARTIN LEE

*Stanford Linear Accelerator Center  
Stanford University, Stanford, CA 94309*

YADUNATH ZAMBRE

*2140 Hanover Street, Palo Alto, CA 94306*

**Abstract**

A new method to operate machine model and beam simulation programs for accelerator control has been developed. Existing methods, although cumbersome, have been used in control systems for commissioning and operation of many machines. We developed GENI, a generalized graphical interface to these programs for model-based control. This "object-oriented"-like environment is described and some typical applications are presented.

*Presented at the Accelerator Control: International Conference on  
Accelerator and Large Experimental Physics Control Systems,  
Vancouver, Canada, October 30–November 3, 1989.*

---

\* Work supported by Department of Energy contract DE-AC03-76SF00515.

## 1. Introduction

Machine modeling and beam simulation codes are the basis of a model-based accelerator control system. Some of these codes were developed for machine lattice design study and others for operation and commissioning, and thus are operated and controlled very differently. There is a great need to develop a generalized graphical environment to operate all of these codes and this need motivated the development of GENI, a General Interface.

GENI provides the user with a color graphics environment comprising a variety of objects such as windows, plots, tables, menus, and buttons. These objects are combined in a unique way to give the user maximum information and control for beamline analysis and design study. It reduces the time-consuming task of learning how to operate a simulation code to the simple task of learning how to play a video game.

We have applied this interface to several modeling codes, including COMFORT [1], a lattice modeling code, and RESOLVE [2], a trajectory error analysis and simulation code. In all cases, we have utilized the underlying object-oriented [3] structure of GENI and built on this to perform the specific functions as required. The use of GENI to operate COMFORT to make changes in the machine functions will be described in this paper to illustrate the ease of use of this graphical environment.

## 2. Existing model interfaces

In general, most of the existing interfaces of the beam optics codes can be described as having four input sections and one output section. The sections are:

- (1) The beamline element definitions,
- (2) The beamline definitions,
- (3) Function select options,

- (4) Output select options, and
- (5) The output in tables and plots.

The first two sections are standard among all codes since they use the MAD [4] language to describe the beamline elements and to define the beamline itself. The differences generally lie in the control and output statements. For example, one code may have control statements for twiss parameter fitting and another for orbit correction. In fact, even if two codes do perform the same function, invariably their control statements will differ. This is the impasse that GENI can be used to overcome.

### **3. New model interface**

The new model interface, GENI, does contain the five sections of the existing methods. The information contained in sections (1) and (2) is obtained from the target simulation code. The remaining three sections are implemented by the user with the graphical interface. GENI presents I/O information and control to the user in its own environment which has been carefully designed for optimal performance. This new environment contains the same number of sections. They are renamed as:

- (1) Input Display window,
- (2) Element Map,
- (3) Input Control menus,
- (4) Output Control buttons, and
- (5) Output Display window.

Figure 1 is a schematic diagram showing the layout of the five sections in GENI. The Element Map, located at the center of the screen, is used to provide the necessary connections to the other sections. This layout allows all five sections to be encapsulated

and therefore the operations only involve message passing among themselves. The encapsulation of each section provides an object-oriented-like environment that makes the integration of GENI to different codes simple.

### *3.1. Interface detail*

Since the Element Map is the focal point of this design, it will be described first. The Element Map displays icons of beamline elements in the order in which they appear in the beamline. Scroll arrows are provided to move the map along the beamline since the number of elements that can be displayed at one time is limited by the workstation graphics and is generally less than the total number of elements in the beamline.

For convenience, the Input Display window is subdivided into three regions. The parameter values that are not changeable, such as the name, type, and location of an element, are displayed in the left region. The center region is used to display the changeable parameter values, such as the field strength for a quadrupole magnet. The right region is used to display the value of alignment errors, such as displacement and rotational misalignments.

The Element Map interacts with the Input Display window by showing parameter values associated with a particular element in the map. This can be done by moving the mouse pointer near the icon in the Element Map. Also, when the pointer is on an element icon a marker will appear in the Output Display window to show the position of this element on the plot. With these features, the user is presented with all the relevant information about a particular element, including its position along the beamline, its values on the plots, and its parameter values, all at the same time. A schematic representation of these connections is shown in fig. 2.

The Input Display window has been designed to perform one of GENI's most innovative and powerful functions. By clicking the pointer on an element icon in the Element

Map, both the center region and the right region of the Input Display window become active regions by converting changeable parameters into buttons. The user can click a button to open a menu for making changes to the element's parameters. In addition, the user can see the effects due to these changes in the Output Display window. Figure 3 shows a more detailed diagram of the Input Display window for individual element function control.

In general, every element parameter can be altered in two ways, namely, simulating or fitting. Simulate is a manual operation to instruct the code to compute the effects on the beam produced by the change in parameter values. Fitting is an automatic operation to instruct the code to change the element parameter value analytically to obtain some desired change on the beam. The combined use of these features provides the user with an interactive environment for rapid analysis and methodical algorithm development.

The remaining regions are essentially the control areas consisting of menus and buttons. The Input Control window is a menu system for function control of the entire simulation session. Controllable options are items such as defining simulation regions, setting the order of approximation, invoking the simulation program, and showing the current state of simulation. The Output Control window consists of buttons that manipulate the Output Display region. The combination of the input and output control windows provides a function-oriented interface that can be used to replace the interface for modeling and simulation codes. The operation of COMFORT will be described to show some of the above features in the next section.

#### **4. Application to COMFORT**

COMFORT is a machine function simulation and fitting code for storage ring design and control applications. Its input file consists of both beamline element and beamline definitions in the standard MAD format. Like most other codes, it has its unique input

data format for function and output control. Figure 4 shows a typical view of the GENI environment for COMFORT.

This figure shows the state of simulation after the user changed the strength of a quadrupole magnet and displayed the corresponding change in a machine function. This change was made by bringing the pointer near the icon of a quadrupole element and clicking the mouse. This action instructed GENI to alter the color of this icon in the `element_map` (making it an active element) and to change the `K1` parameter (the strength of this element) into a button. Clicking this button opened a menu which allowed the user to enter a new strength (see fig. 3). This new value was then displayed in the Input Display window as an Actual Value.

To see the effects of this change, the user clicks the Simulate option on the Input Control menu and the result is displayed according to the Output Control selected in the Output Display. It is possible to display a plot of any machine function (*Beta*, *Alpha*, *Eta* or *Eta'*). In addition, the user can create a another display window with the New Window button to see two or more plots at the same time. All these features provide the user with a comfortable interactive environment to operate COMFORT.

## 5. Integration with other beam codes

GENI can be viewed as a framework which interacts with machine modeling and beam simulation codes as follows:

- (1) Extract and display information,
- (2) Change parameter values, and
- (3) Perform simulation and fitting.

Since GENI is object-oriented in nature, it can be integrated with other codes by implementing only a few message-passing routines to perform these functions. Extract-

ing information from the target code is generally the most laborious part. Changing parameter values and performing fitting or simulation is relatively easy. With a handful of routines, GENI can be quickly integrated with any modeling code for discrete element beamlines.

## **6. Portability to workstations**

Although GENI was developed on a DEC VAXSTATION 200 using UIS graphics, it was designed with the intent to port this graphical interface to other workstations. With that in mind, a kernel was essentially written to translate GENI's graphics commands to the device-dependent routines. The most time-consuming work will be to implement the interrupt mechanism for menu and button control since all other tasks, such as draw a line or open a window, are easily implemented. It has been estimated that the time required to implement GENI on a Macintosh is one week.

## **7. The NEXT implementation**

A new application is to implement GENI in a portable trajectory analysis and correction system that can be used in any control system, or as a stand alone program. This new system, THE WORKS ( THE WORKstation Solution ), is being developed using a dedicated NEXT computer. Figure 5 is a schematic diagram showing the elements in the THE WORKS.

The upper half of the diagram shows how THE WORKS can be used to develop rules for expert systems or to train neural networks for adaptive control. After they have been developed and trained, the lower half of the diagram shows how they can be used to analyze or control the beam trajectory data automatically.

Presently, the center three elements in the diagram have been completed. To transport THE WORKS to a different control system, only the Interface to Database element needs to be customized.

THE WORKS is currently being implemented to operate SPEAR. At the same time, work is in progress to develop rules and neural networks to automate some of the manual operations.

## 8. Summary

The use of a model-based program to commission and control a complex accelerator system has led to the recent success in the operation of SLC. The work described in this paper represents only the first step toward the development of a model-based control system that will be needed to commission and operate future accelerator systems more effectively and effortlessly.

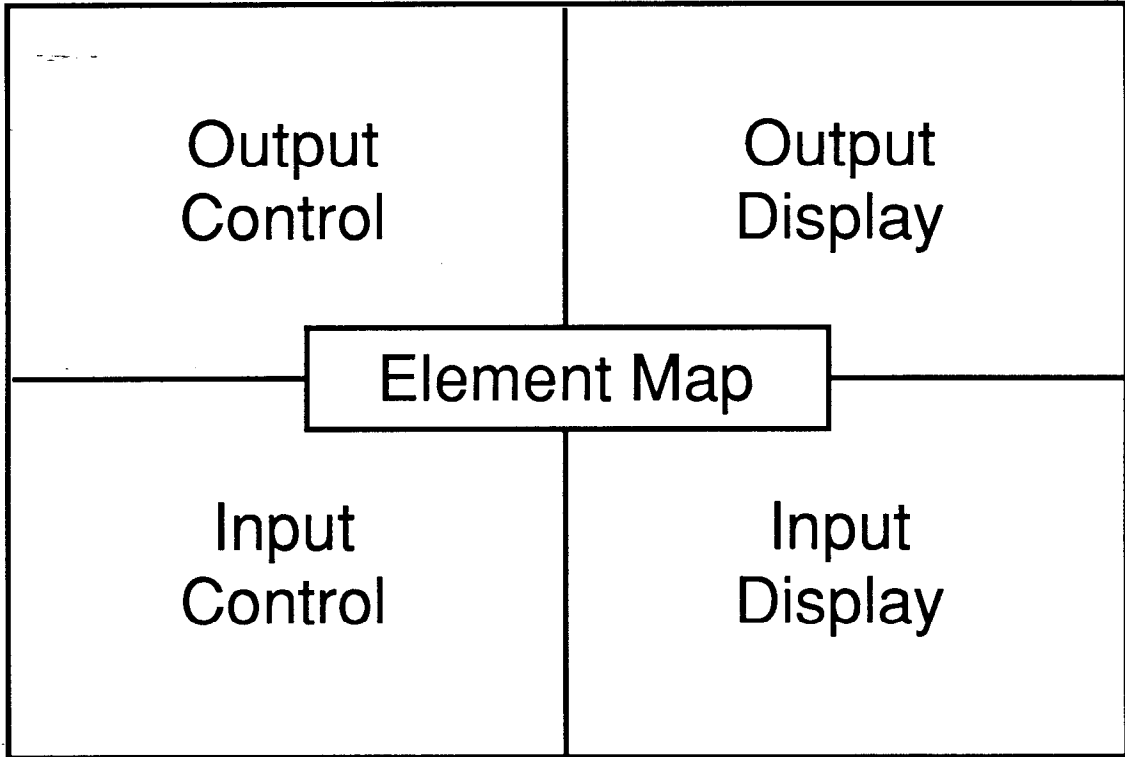
## References

- [1] M. D. Woodley, M. J. Lee, J. Jaeger, and A. S. King, COMFORT, Control Of Machine Functions OR Transport Systems, in: Proc. 1983 Particle Accelerator Conference, Santa Fe, New Mexico, March 1983.
- [2] Zambre, Lee, and Kleban, private communications (1988).
- [3] B. J. Cox, Object-Oriented Programming, An Evolutionary Approach (Addison-Wesley, Reading, Massachusetts, 1986).
- [4] F. C. Iselin and J. Niederer, The MAD (Method Accelerator Design) Program, Version 7.2, User's Reference Manual, CERN/LEP-TH/88-38 (1988).



## Figure Captions

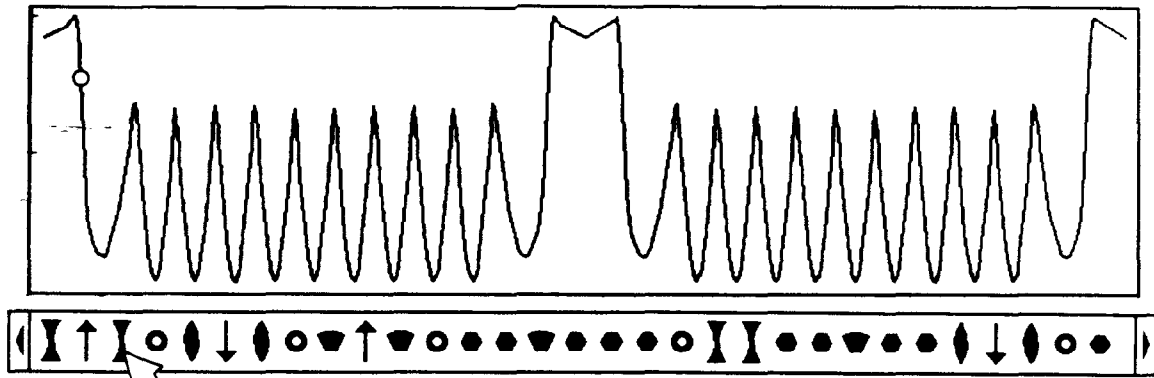
1. A schematic of the five sections that comprise the interface to modeling and simulation codes.
2. A partial view of the GENI interface to show the connections of the Element map to the Output and Input Display.
3. A view of the Input Display window after the user clicks the mouse pointer on an element icon in the Element map. Changeable items appear with boxes surrounding them, making them active for user actions.
4. A typical view of the GENI environment showing the effects on the Beta function after the user changes the strength of a quadrupole magnet.
5. A schematic diagram of THE WORKS to show how Neural Networks, Expert Systems, Modeling and Simulation codes, and a Graphical User Interface can be combined into a single system.



10-89

6504A4

Fig. 1



Element Information		Parameters:			Alignment Changes
		Design	Actual	Units	
NAME	QFIH	LENGTH	0.08	m	dx (m)
TYPE	QUAD	K1	-9.74	1/m**2	dy (m)
INDEX	5				dz (m)
Dist (m)	1.19				rx (rad)
					ry (rad)
					rz (rad)

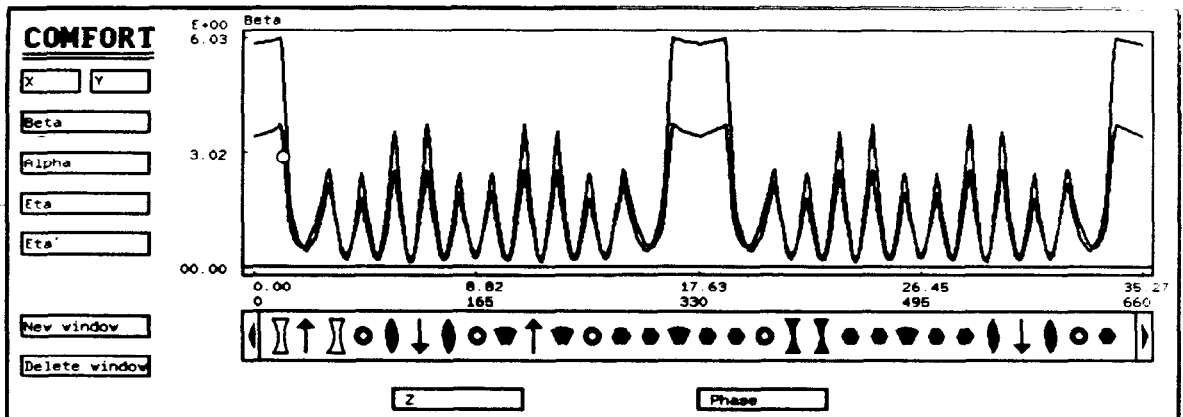
10-89

6504A2

Fig. 2

Element Information		Parameters:			Alignment Changes	
		Design	Actual	Units		
NAME	QFIH	LENGTH	0.08	m	dx (m)	
TYPE	QUAD		-9.74	1/m**2	dy (m)	
INDEX	5				dz (m)	
Dist (m)	1.19				rx (rad)	
					ry (rad)	
		<div style="border: 1px solid black; padding: 2px;">           On for fitting            Set to design value            Set to new value            Exit         </div>			<div style="border: 1px solid black; padding: 2px;">           Enter value, (x to exit)         </div>	
		<div style="border: 1px solid black; padding: 2px;">           ‡ -9.4_         </div>			Unfreeze	

Fig. 3

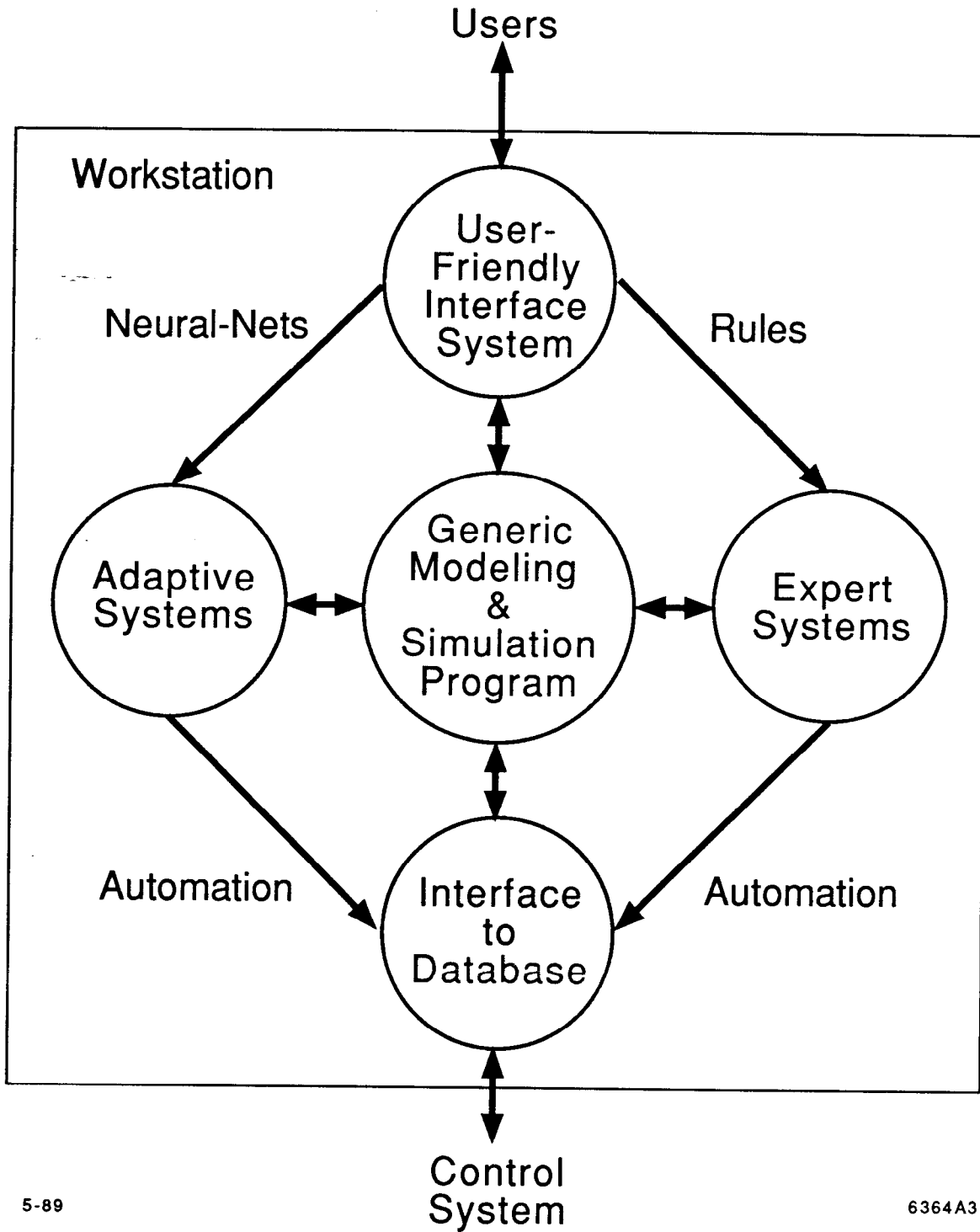


Element Information		Parameters:				Alignment Changes	
NAME	QFIN	LENGTH	Design	Actual	Units	dx (m)	
	QUAD	K1	-9.74	-9.40	1/m**2	dy (m)	
INDEX	5					dz (m)	
Dist (m)	1.19					rx (rad)	
						ry (rad)	
						rz (rad)	

- Files
- Simulation Region
- Clear fit variables
- Simulate
- Fit
- Exit

Fig. 4

# THE WORKS



5-89

6364A3

Fig. 5