PROTOTYPE DEVELOPMENT OF A BEAM LINE EXPERT SYSTEM*

SCOTT H. CLEARWATER†
Los Alamos National Laboratory, Los Alamos, New Mexico 87545

MARTIN J. LEE
Stanford Linear Accelerator Center, Stanford, California 94305

Introduction

Modern particle accelerator facilities are large and complex machines consisting of a large number of beam-lines. One of the tasks of running a facility is beam-line start-up, which occurs whenever the machine is upgraded. In an effort to improve start-up efficiency, separate beam-lines are commissioned concurrently with the use of beam-trajectory correction and profile-matching programs. In order to reduce the software effort needed for start-up, the code is written in a generalized form so that the same code can be adapted to different parts of the machine. In spite of this effort, there is still a shortage of the expertise needed to plan and analyze commissioning experiments. We believe that an expert system can be used to alleviate this problem of lack of expertise.

The Quandary

In the past, and in some cases the present, the rush to perform experiments at accelerator facilities has inhibited the development of automated and systematic procedures that are useful for the commissioning and start-up of the facility. This is true despite the fact that particle accelerator and storage ring systems are becoming larger and more complex. Even the cost of electrical power, typically several thousands of dollars per hour, is a major financial concern facing every laboratory in the world today. The success of experiments performed at a facility depends largely on how much data can be collected during the time the experiment is scheduled to run. Thus, the success of the entire accelerator facility rests mainly on how well and how often the beam is delivered to the experimenters.

The start-up problems become even more important because extended shut-downs often occur several times per year and so-called commissioning tasks are underway almost continuously as the machine is upgraded or the elements are readjusted. Therefore, it is important to find an effective and efficient way of operating and controlling these systems as well as minimizing the time for commissioning and starting up after a shut-down. In particular, programs for correcting errors in a beam or a machine are now used in the operation of particle accelerator facilities throughout the world. Underlying these programs are physical models of beam transport. Because many of the programs are often not automated, their use is limited, to the few people who know how to set up, execute, and interpret them.

An Automated Intelligent Solution

As an experiment to improve beam-line performance and to solve some of the start-up problems, we have begun to automate the use of beam and machine correction programs into a unified "expert system".[1] An expert system is a computer program that can perform a task requiring human expertise. Typical characteristics of expert systems are their use of "qualitative" processing, great flexibility, and ability to handle very complex problems. Another feature is the relative ease of upgrading and debugging. Also, an expert system means that the expertise is always available. In addition, we have also developed a beam-line simulation program to assist in the codification and formalization of the beam-line optimization techniques. The simulation has been extremely valuable both as a time saver of real beam time and as an efficient means of acquiring knowledge for the expert system.

In this paper, we present a paradigm for beam-line commissioning and start-up using a hybrid expert system. A hybrid expert system combines conventional expert system technology, physical models, and sophisticated mathematical optimization techniques into an integrated system that utilizes the best of the quantitative and qualitative powers of computers into a single unit that is much more powerful than any of the individual pieces alone. The next section discusses beam-correction programs that correct the *symptoms* of a beam-line problem during routine operation. Also discussed are machine-correction schemes that attempt to locate the *causes* of beam-line problems during commissioning. Next, the uses and advantages of beam-line simulation are discussed in the context of learning tools for the development of beam-correction and machine-correction schemes. Finally, we discuss the prototype development of an expert system that integrates beam-correction, machine-correction, and simulation programs into a unified package.

Machine Modeling Programs

Today, in order to optimize the useful beam-time, most of the existing accelerator and storage ring systems are operated under computer control. Mathematical models are used to set the strength of the beam-line elements. Using these models, the strength of the beam-line corrector magnets are also calculated so as to minimize the effects due to errors in the beam-line elements. The programs used to correct the effects on the *beam* due to errors in the *machine* will be referred to as beam-correction programs.

The problems of commissioning an accelerator or storage ring system are different from those of its routine operation. During commissioning, it is important to find the causes of the errors in the beam-line elements as quickly as possible in order to maximize precious beam time for the experimenters. Unless the magnitudes of the errors are sufficiently small, the solution of the beam-correction program cannot be implemented because the strength of the correction elements is limited. Thus, new programs for finding the errors in the beam-line elements are needed for commissioning.[2] The use of these new machine-correction programs allows the users to determine the nature of the problems. Once the causes of the errors are found, they can be corrected physically in the beam-line, *e.g.* by re-adjustment of the beam elements or compensated for mathematically in the model.[3]

Beam Simulation Program

For a large and complex system, the commissioning process can be very costly in terms of man power and electrical power. Past experience has shown that a considerable amount of time and effort has been spent in commissioning beam-lines using untested programs. Therefore, before modeling programs can be used for routine operation of an accelerator system, they must be thoroughly tested. One way to develop and test any

* Work performed under the auspices of the U.S. Department of Energy and supported by the U.S. Army Strategic Defense Command and by the Department of Energy under contract DE-AC03-76SF00515.

† Presently visiting scholar at the Knowledge Systems Laboratory, Stanford University, Palo Alto, California 94304.

new correction scheme is to use a beam-line simulator. A beam-line simulator consists of a model of the *machine* to calculate the effects on the *beam* introduced by given errors in the beam-line elements. In a simulator, a model of the machine can be represented by the machine functions computed by the lattice design program. The effects on the beam can then be calculated from the values of the machine functions. In particular, our simulator uses the COMFORT[4] program.

The use of a beam-linesimulator has many advantages over using the real beam-line. Unlike the real system, which breaks down often, a simulator is usable whenever the computer is available so that a large variety of procedures and strategies can be tested in a short time. The comparison of the results can be done easily, resulting in a large saving of time and effort. Additionally, a beam-line simulator can also be useful as a training tool for operators, much like flight simulators have been used to help train airplane pilots for some time.

Simulator Test Case

A procedure for finding alignment errors[5] in beam-line elements hasbeen developed and studied using a beam-line simulator. In this test case, a COMFORT dataset is used to define a model of the machine. In addition to the actual elements in the beam-line, a pseudo thin-lens dipole corrector is assumed to be located at every element. The alignment error of an element is given by the strength of the respective pseudo-corrector. The beam position at each monitor in relation to the *given* alignment error at each selected element is then calculated. The resolution error at each monitor is generated randomly and folded into the beam position to give the simulated trajectory. Then, the simulated trajectory values are analyzed using a program to find the *fitted* value of the pseudo-correctors which produce the same trajectory as on the simulated beam. The response of the program is considered a success if the *fitted* values of the pseudo-corrector strengths agree with the *given* alignment errors.

Several methods have been tried in the machine-correcting program. Two of the successful methods operate under complementary assumptions. The first technique is a global method that finds a solution assuming that all of the elements are misaligned. The program uses a sophisticated constrained nonlinear optimization program, NPSLAC to vary the strengths of the pseudo-correctors at each element in order to minimize the difference between the trajectory of the model and the trajectory from the machine. NPSLAC allows bounds to be placed on the fitted variables which means that the fitted pseudo-corrector strengths returned by the program will be within reasonable limits corresponding realistic mis-alignments on the actual machine. The second technique is a local method which assumes that only a few of the elements are mis-aligned.

Many experiments have been performed with different mis-alignment situations using the beam-line machine simulator. A great deal of knowledge has been gained by doing these experiments. For example, the rate of success between the global and local methods has been compared. Also, we have found that the global method can be useful for checking the beam-line for large alignment errors independent of the nature of the alignment errors. However, for large alignment errors that are localized at a few elements, the local method has been found to be more useful than the global method. Practical rules for identifying these elements have been developed. The limitations of both methods has been explored.

Automated Beam Line Expert System

ABLE is the Automated Beam Line Expert and is currently being developed at the Stanford Knowledge Systems Laboratory. ABLE combines the codified knowledge, reasoning, modeling, and optimization techniques we have used into an integrated system. In this way, ABLE can provide the best of both worlds by combining numerical algorithmic procedures with symbolic reasoning akin to humans. The importance of the automation that ABLE provides is that the tasks are performed systematically and consistently which could result in increased efficiency of a facility. ABLE's purpose is to design and analyze beam-line experiments using simulated or actual beam-line data and various machine-correction programs as described in the previous sections.

In developing ABLE, we are using the standard sorts of methods used in expert systems such as frame-like data structures, use of rules and logic, forward and backward chaining, etc. Another issue in expert system performance is time response. In the case of commissioning, the time constraints on finding a solution are quite loose because much of the time is spent analyzing data and thinking about what should be done next. However, in other cases, the environment is rapidly changing so that the expert system should have a "real-time" capability.

Traditionally expert systems have relied on symbol manipulation rather than numerical computation. But, most of the data available to a beam-line expert system are numerical and at least some of the numerical data will need to be converted to symbolic form. One representation is to convert the continuous real number valued data into several discrete values.

The ABLE Prototype

We have begun development of the ABLE prototype, to find element mis-alignments in a simulated beam-line as described in the section on the beam simulation program. This prototype is a test case for automating beam-line error-finding tasks. We will now describe some of the salient features of the ABLE prototype.

ABLE uses its knowledge of the beam-line to decide the best experiments to perform. For example, ABLE will not perform experiments that would look for an error downstream from where it believes a problem exists. ABLE compares the results of the various experiments and can discard or perform new experiments in order to refine its hypothesis about the problem. The right portion of Fig. 1 shows a flowchart of the ABLE process.

It should be noted that these methods are still evolving and may change significantly as more studies are performed using the simulated beam-line. The computer software we are using has a sophisticated expert system building environment that permits rapid prototyping. The computer is able to run both numerical (FORTRAN) and symbolic (LISP) programs. This capability reduces many of the problems that occur if one relies on separate computers for numerical and symbolic processing. The use of an expert system building shell allows modifications and tests to be made very quickly.

In the future phases of the ABLE project, we hope to incorporate new techniques[6] into the expert system. We also will study the effects of noisy data on the error-finding techniques and on the interpretation of the experiments. Since ABLE is designed to be flexible enough to incorporate new strategies without extensive rewriting of code, the inclusion of beam-correcting schemes is a natural and easy extension toward the automation of control and operation of a beam-line facility.

Summary

What we have presented represents a first step in what we believe to be the proper direction toward solving complex accelerator control and commissioning problems. In building ABLE, we have demonstrated that the use of a beam-line simulator can greatly reduce the costly task of "knowledge engineering" needed to extract the expertise from the experts. Using the simulator, the expert system programmer can test, and even develop, new techniques with much less than the usual amount of human expert interviewing. Also, the advent

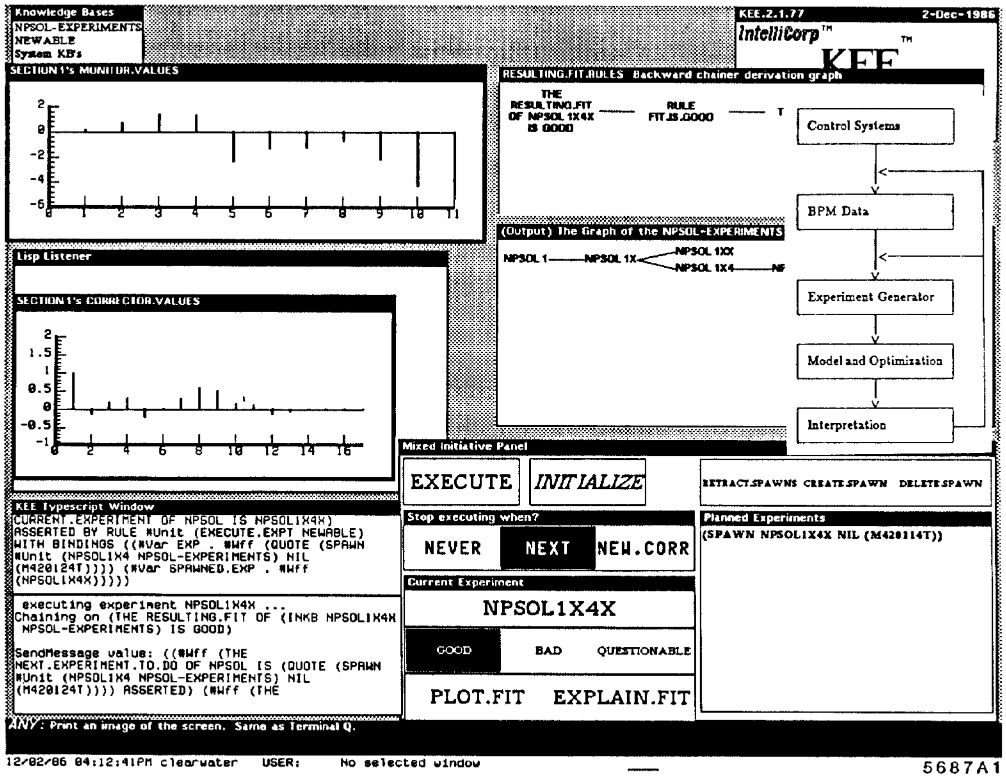

Fig. 1. A typical ABLE display screen. The middle left of the screen shows a pseudo-corrector plot and the corresponding BPM data. The right part of the screen shows a flowchart of the ABLE process. The other parts of the screen are for running and controlling ABLE.

of the simulator has led directly to the development of a new alignment-error finding technique. For the problems tested so far, our methods have produced results that are as good as the expert's.

Further research is needed to expand the domain of expertise of our expert system to include more of the capabilities needed for running an accelerator facility. Finally, the joining of three potent techniques, an expert system, a well-understood physical model, and sophisticated mathematical optimizations, into a single hybrid expert system we expect will give an accelerator facility a very powerful tool with which to deal with its very complex problems.

Acknowledgements

We wish to thank Margaret Wright of the Stanford Systems Optimization Laboratory for helping us to use NPSLAC. We also wish to thank Larry Selig for coding the ABLE prototype, Stephen Kleban for assisting in the development of the machine-error programs, and Robert Engelmore for providing general guidance in the development of ABLE at the Stanford Knowledge Systems Laboratory.

References

1. F. Hayes-Roth, D. Waterman and D. Lenat, eds., *Building Expert Systems* (Addison-Wesley, Massachusetts, 1983).
2. M. J. Lee, S. H. Clearwater and S. D. Kleban, "Error-finding and Error-correcting Methods for the Start-up of the SLC," in Proceedings of the 1987 Particle Accelerator Conference.
3. J. Jaeger, M. Lee, M. Woodley and J. Delahaye, "Modeling of the SLC Damping Ring," SLAC-PUB-3408, August 1984.
4. M. D. Woodley, M. J. Lee, J. Jaeger and A. S. King, "COMFORT, Control Of Machine Functions OR Transport Systems," presented at the 1983 Particle Accelerator Conference, Santa Fe, New Mexico March 21-23 1983.
5. M. J. Lee, S. H. Clearwater and S. D. Kleban, "Error-finding and Error-correcting Methods for the Start-up of the SLC," in Proceedings of the 1987 Particle Accelerator Conference.
6. M. Lee, S. Clearwater, E. Theil, V. Jacobson and V. Paxson, in the Proceedings of the 1987 Particle Accelerator Conference.