10
5-29-91 (1)

SLAC-PUB--5470

DE91 012388

USE OF DIGITAL CONTROL THEORY STATE SPACE FORMALISM FOR FEEDBACK AT SLC*

T. Himel, L. Hendrickson, F. Rouse,[a] and H. Shoaee

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

Abstract

The algorithms used in the database-driven SLC fast-feedback system are based on the state space formalism of digital control theory. These are implemented as a set of matrix equations which use a Kalman filter to estimate a vector of states from a vector of measurements, and then apply a gain matrix to determine the actuator settings from the state vector. The matrices used in the calculation are derived offline using Linear Quadratic Gaussian minimization. For a given noise spectrum, this procedure minimizes the rms of the states (e.g., the position or energy of the beam). The offline program also allows simulation of the loop's response to arbitrary inputs, and calculates its frequency response.

INTRODUCTION

The SLAC Linear Collider (SLC) is a novel accelerator designed to produce e^+e^- collisions at center-of-mass energies up to 100 GeV, i.e., around the mass of the neutral intermediate vector boson Z^0. The collisions occur between bunches of electrons and positrons, which are produced, accelerated, collided, and dumped at a maximum rate of 120 Hz. When the feedback project described here was started, the SLC had fast-feedback loops that stabilized both the energy of the machine [1] and the orbit, through a set of collimators near the end of the linear accelerator. These feedback loops were essential to the operation of the SLC.

We have designed a new system that replaces the current software with generic, database-driven software [2]. We rely on the SLC database to specify each different loop. This is possible because the action of any feedback loop can be cast into a series of matrix equations in the formalism of digital control theory [3].

This paper briefly introduces the state space formalism of modern control theory, and then describes how it has been applied to the SLC feedback system. Some of the tradeoffs involved in the design of a loop are discussed.

STATE SPACE FORMALISM

The state space formalism provides a simple, elegant method to describe the dynamics of a linear system in a single matrix equation. We now give that equation, and an example of its application to modeling the response of a section of the SLC. Define the following vectors:

- **y** is a vector of outputs. For our example elements of this vector are readings of beam positions from Beam Position Monitors (BPMs).
- **x** is a vector of states. A typical state is the position, the angle, or the energy of the beam at a point. Note that these are not necessarily directly measurable with a sensor. Other states are the angular kicks given to the beam by devices (typically magnets) in the beam. These may be devices directly controlled by the feedback loop, or devices upstream of the loop whose perturbations must be corrected by the loop. The state vector should completely describe the system.
- **u** is a vector of inputs. A typical input is the current requested for a magnet.

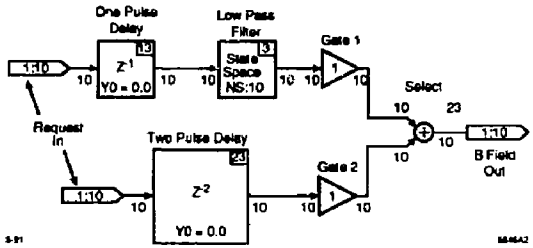

Figure 1. Diagram of actuator dynamics. Normally, either gate *1* has a gain of 1 and gate *2* a gain of 0 or vice versa. In the first case, the actuator is assumed to have an RC time constant due to filtering in the power supply. In the second case, the response is modeled as a delay of two beam pulses. This accounts for computation time of the feedback loop and other minor delays.

These vectors are then related with a general first-order matrix equation using the four matrices **A**, **B**, **C**, and **D**:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) .$$

These two equations can be combined into a single one by defining the system matrix, **S**:

$$\mathbf{S} = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix} .$$

This gives the continuous state space equation:

$$\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{y}(t) \end{pmatrix} = \mathbf{S} \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{pmatrix} .$$

In the above equation, **x**, **y**, and **u** are all considered to be continuous functions of time. The equation below is used for the discrete (sampled) case. The subscript k indicates the k^{th} sample:

$$\begin{pmatrix} \mathbf{x}_{k+1} \\ \mathbf{y}_k \end{pmatrix} = \mathbf{S} \begin{pmatrix} \mathbf{x}_k \\ \mathbf{u}_k \end{pmatrix} .$$

There are multiple advantages to using this formalism. It is a standard formalism, so there are books and computer tools that can be used to aid in the development. Secondly, the software tends to be general and streamlined.

APPLICATION TO THE SLC

Modern control theory was developed to handle dynamical systems. A satellite attitude control system, for example, involves an object with a finite moment of inertia, and its state (spin velocity) at a certain time is related by a differential equation to its state at a

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

[a]Present address: University of California at Davis.

Presented at the IEEE Particle Accelerator Conference, San Francisco, CA, May 6-9, 1991

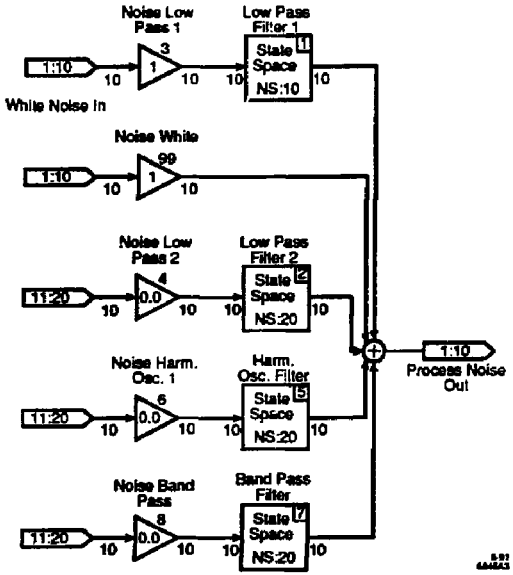

Figure 2. Diagram of beam noise dynamics. Normally only two or three of the gain blocks on the left have nonzero gains. The time constants of the low-pass filters, and the frequency and quality factor of the harmonic oscillator filter, can be adjusted to get the modeled noise spectrum to match the measured one.

previous time. At first glance, the SLC accelerator has no dynamics, no *inertia*. There is a bit of dynamics in the actuators, typically correction magnets, that are used by feedback to steer the beam. The model used for this is shown in Fig. 1. However, the typical magnet is fast on the scale of the 1/120 second period between beam pulses, so this dynamics is not essential to the problem.

The important part of modelling the dynamics of the system is the description of what is causing the beam to move. For example, there may be an oscillating or slowly varying upstream magnet power supply, causing the beam to move. The dynamics of the ensemble of all such power supplies must be modeled. In practice, we measure the spectrum of disturbances to the SLC beam, and then try to match that by adjusting model parameters such as low-pass filter time constants and oscillation frequencies. Figure 2 shows the model used for this dynamics of the noise source.

FEEDBACK ALGORITHM

Having described the model of the accelerator in the state space formalism, we can now go on to design the feedback algorithm. For this, the predictor-corrector formalism of optimal control theory [3] is used. The first controller equation is used to estimate the present value of the state vector:

$$\hat{\mathbf{x}}_{k+1} = \Phi\hat{\mathbf{x}}_k + \Gamma\mathbf{u} + \mathbf{L}(\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}_k)\,,$$

where

- $\hat{\mathbf{x}}_k$ is the estimate of the state vector on the k^{th} pulse.
- Φ is the system matrix, and describes the dynamics of the accelerator model.

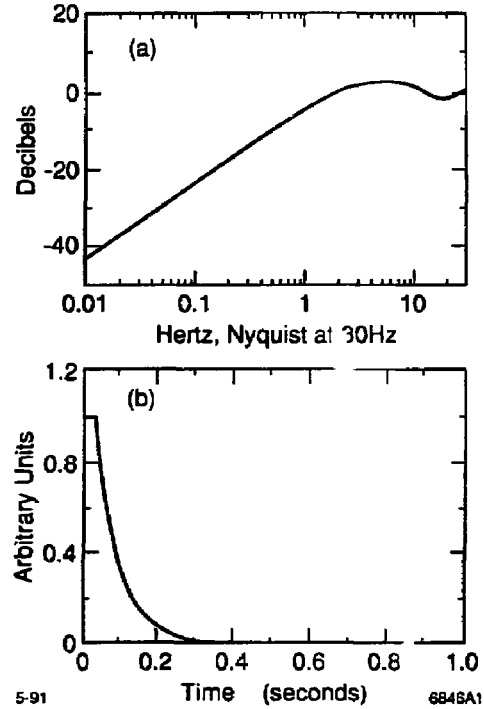

Figure 3. Simulation results: (a) shows frequency response for a typical loop. (b) shows the response to a step change in the incoming states. Typical states recover in 0.1 sec (for a 60 Hz sample rate).

- Γ is the control input matrix. It describes how changes in the actuators should affect the state.
- $\mathbf{H}$ is the output matrix. It maps the state vector to the output vector. That is, given an estimate of the states, it gives an estimate of what the sensors should read.
- $\mathbf{L}$ is the Kalman filter matrix. Given an error on the estimate of the sensor readings, it applies a correction term to the estimate of the state vector.

The matrices Φ, Γ, and $\mathbf{H}$ are obtained from the model of the accelerator. The $\mathbf{L}$ matrix is derived from the other matrices, and is designed (via the Linear Quadratic Gaussian method) to minimize the rms error on the estimate of the state.

The second controller equation calculates the actuator settings from the estimate of the state vector.

$$\mathbf{u}_{k+1} = \mathbf{K}\hat{\mathbf{x}}_{k+1} + \mathbf{N}\mathbf{r}$$

where

- $\mathbf{r}$ is the reference vector that contains set points for the loop.
- $\mathbf{N}$ is the controller-reference input matrix. It maps the reference vector to actuator settings and is directly derivable from the model of the accelerator.
- $\mathbf{K}$ is the gain matrix. It is derived in a manner similar to $\mathbf{L}$. It is designed to minimize the rms of selected state vector elements.

DESIGN TRADEOFFS

There are several design goals for a feedback loop. It should minimize the rms of the states it is trying to control, have a good response to a step change in the incoming beam, have a good frequency response, have a good DC bias rejection, and continue functioning well even if the accelerator is slightly different than the model used by feedback. These design goals are conflicting, so tradeoffs must be made.

The tuning of feedback is done using an offline simulation program. This program is built on top of the Matrix$_x$ control system CAD package written by Integrated Systems Incorporated. All the figures in this paper are outputs from this simulation program. The program takes parameters (such as filter cutoff frequencies) from the on-line database, calculates all the matrices needed by the feedback loop, saves them for use by the loop, and then calculates and plots the frequency response, the response to a step function, and so on. The user can vary the parameters and rerun the simulation until the desired response functions are obtained. Two examples of these plots for a typical loop setup are shown in Fig. 3.

CONCLUSIONS

At present there are seven feedback loops running at the SLC that are designed and implemented using the state space formalism of digital control theory described above. These loops control a total of 52 beam parameters. They work quite well, and there is demand for more. We expect to have about 12 feedback loops operating by the end of the year.

As more loops have been designed and implemented, the power and generality of a database-driven feedback system based on the vectors and matrices of modern control theory have become clear. Implementing new loops requires little or no new on-line software.

ACKNOWLEDGMENTS

We thank John Zicker for originating the concept of applying modern control theory to our feedback problem, and we thank Robert McEwen of Integrated Systems Incorporated for his help in applying that theory to our problem.

REFERENCES

[1] G.S. Abrams et al., "Fast Energy and Energy Spectrum Feedback in the SLC Linac," Proc. 1987 IEEE Particle Accelerator Conf., Washington, D.C, 1987.

[2] F.R. Rouse et al., "General, Database-Driven Fast-Feedback System for the Stanford Linear Collider," this conference.

[3] Gene F. Franklin and J. David Powell, Digital Control of Dynamic Systems, (Addison-Wesley, 1980).