Six-Dimensional Modeling of Coherent Bunch Instabilities and Related Feedback Systems in Storage Rings with Power-Series Maps for the Lattice*

J. Bengtsson
Lawrence Berkeley Laboratory, Berkeley, CA 94720
D. Briggs
Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309
M. Meddahi
now at CERN-SL, CH-1211 Geneva 23, Switzerland

Abstract

We have developed 6-dimensional phase-space code that tracks macroparticles for the study of coherent bunch instabilities and related feedback systems. The model is based on power-series maps to represent the lattice, and allows for straightforward inclusion of effects such as amplitude dependent tune shift, chromaticity, synchrotron oscillations, and synchrotron radiation. It simulates long range wake fields such as resistive-wall effects as well as the higher order modes in cavities. The model has served to study the dynamics relevant to the transverse feedback system currently being commissioned for the Advanced Light Source (ALS). Current work integrates earlier versions into a modular system that includes models for transverse and longitudinal feedback systems. It is designed to provide a modular approach to the dynamics and diagnostics, allowing a user to modify the model of a storage ring at run-time without recompilation.

INTRODUCTION

The dynamical model has been developed with emphasis on computer implementation. This approach allows one in general to consider a fairly complete description of the dynamics, leading to numerical studies. However, such an approach does typically not give the insight into the dynamics that an analytic approach may provide. On the other hand, the numerical approach has generally considerably larger range of validity (particularly in the nonlinear case). Our point of view is that the two approaches are complementary and have therefore pursued analytical studies in parallel with computer modeling [1].

THE WAKE FIELD MODEL

The dynamical model is based on maps. This allows for modularity of the dynamics and corresponding ease of computer implementation. The one turn map $\mathcal{M}$ can be written

$$\mathcal{M} = \mathcal{M}_{12}\ \mathcal{M}_{23} \cdots \mathcal{M}_{n-1,n}$$

A set of well established, state of the art tools for single particle beam dynamics (Tracy-2, Despot and the DA-library) are used to extract power series maps modeling the lattice. This allows for straightforward inclusion of effects such as amplitude dependent tune shift, chromaticity, synchrotron oscillations, and synchrotron radiation. These maps are used to model the center of charge (rigid bunch) motion of macroparticles. Note however, that the power series representation of the maps are non-symplectic and the model must therefore be carefully calibrated.

The wake fields are modeled by the corresponding wake potentials, implemented as thin kicks. Long range wake fields, like the resistive wall effect, are replaced by an equivalent thin kick obtained by averaging the equations of motion along the lattice. This is systematically done to first order in the strength, by using a Hamiltonian formulation [1]. This formalism also allows for a straightforward generalization to the case of varying beta function along the lattice. Lumped circuit concepts, like impedance, are introduced as the Fourier transform of the wake potential.

In the case of higher order cavity modes, one is dealing with a high Q, narrow band resonator. It follows that the computer model only has to keep track of the amplitude and the phase of the resonance frequency, and the excitations are simply superimposed.

For the resistive wall case, the wake field is only damped as $1/\sqrt{s}$. The tracking code must therefore be able to store each excitation for some duration. Note that the neccessary number of stored excitations, determined by the number of bunches and the duration of excitations, will drastically affect the performance.

VERIFICATION OF THE COMPUTER MODEL

An important aspect of the computer implementation has been its numerical verification. This is far from a nontrivial task due to the complexety of the dynamics (6-d phase space, all bunches may potentially interact through the wake fields, the power-series maps are non-symplectic, active feedback systems, etc.). The numerical simulations have been successfully checked against analytical calculations in situations where the two approaches overlap and by scaling tests (scaling of the quality factor and the impedance). Analytical calculations include: calculation and tracking of eigenmodes, calculation of exponential and linear (for bunches with initial zero amplitude) growth rates in cases where the higher order modes have been detuned onto the sidebands. Tracking based on the power-series maps has been compared with a symplectic integrator (Tracy-2) to determine required expansion order and related limitations on the betatron amplitudes, for relevant time scales.

*This work was supported by the Director, Office of Energy Research, Office of High Energy and Nuclear Physics, High Energy Physics Division, of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098

RELATED FEEDBACK SYSTEMS

The modular system includes models for transverse as well as longitudinal feedback systems. Some preliminary numerical and analytical performance studies have been made recently. System analysis by application of (classical) linear control theory, allows for a systematic pole placement. This approach leads therefore, not only to a clear understanding of the dynamics, but also to insight of both limitations as well as possible improvements of the performance [2]. Fig. 1 and 2 illustrates the difference in performance between a "naive" choice of gain coefficients vs the well known case of deadbeat control. The feedback system is turned on at turn 100 in both cases.

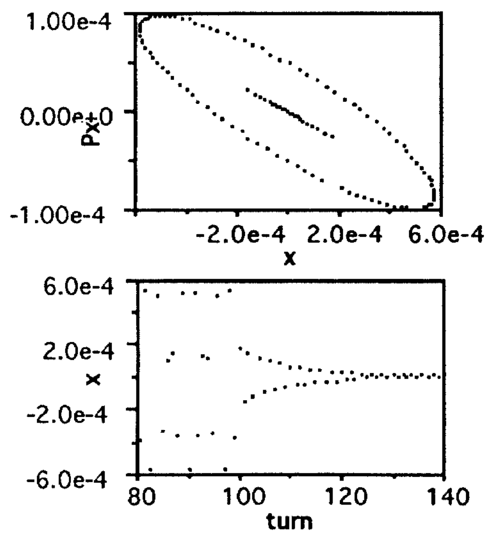

Fig. 1: Damping using a "naive" choice of gain matrix

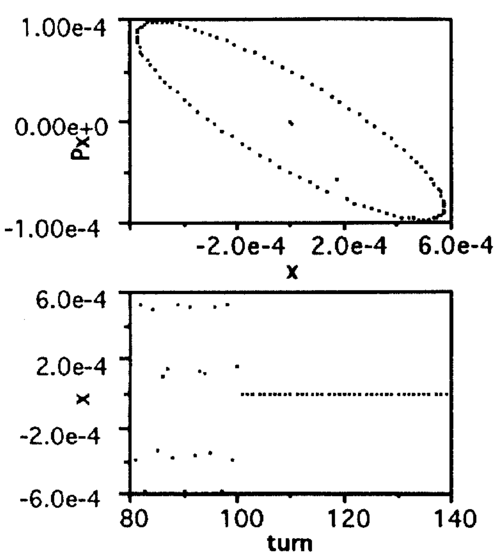

Fig. 2: An example of transverse deadbeat-control feedback

Note that the computer model will be particularly valuable for systematic studies of saturation effects due to limited actuator strengths and nonlinear effects due to nonlinear contributions from e. g. the sextupoles in the lattice. In addition, careful comparisons with experimental data should of course, as always, prove interesting.

FUNCTIONAL OVERVIEW

SimRing simulates the kinematics of macroparticles in a storage ring in the time domain [3]. It belongs to the family of continuous (rather than discrete-event) simulations. The internal representation uses full six-dimensional phase space, and arbitrary charge. The number of macroparticles is arbitrary, up to the harmonic number of the RF frequency for the ring.

SimRing supports a kinematic model comprising a representation of a ring with model elements. The model elements comprise (a) points on the ring representing impedances, sensors, and actuators; and (b) propagation from point to point on the ring. SimRing supports any number of each kind of model element, in arbitrary order. A simple text configuration file determines the order, identities, and properties of the model elements at run time.

The simulation can model an arbitrary number of RF cavities, placed anywhere in the ring, each supporting an arbitrary number of transverse and longitudinal modes. It also models the resistive wall effect (presently, transverse only). This approximation truncates the time-duration of the resistive wall excitations, expressed by the user in units of turns (floating-point).

The simulation models time-domain bunch-by-bunch longitudinal and transverse feedback systems. The sensors and actuators model noise, quantization errors (if desired), saturation, and bandwidth limitations.

The simulation can report to a file the 6-d phase space or the variation of calculated linear invariants of any macroparticle on any turn at any point on the ring. Truncated power-series maps of arbitrary order describe the propagation of macroparticles from one point on the ring to another, up to one full turn.

STRUCTURAL OVERVIEW

SimRing comprises a Controller and an Engine. The Engine manages kinematics of macroparticles and the time-order of interactions of the macroparticles with the model elements. The Controller informs the Engine of what file to use for configuration, how many turns to simulate, and when to save results. Typically, a Graphical User Interface permits the user to exercise Controller functions directly. Alternatively, the user exercises these functions through user-written code.

The Engine establishes a software back plane with a ring topology, and installs in it an arbitrary number of software modules, each representing a model element. The configuration file specifies the model completely. See Fig.-3.

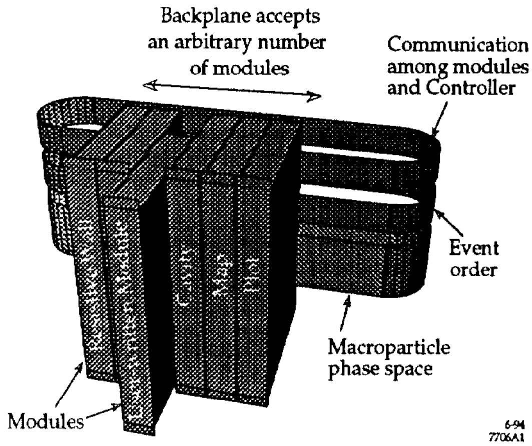

Fig. 3: The Software Back Plane and Model Elements

USER INTERFACE

Since the model elements comprises points on a ring and the connections from point to point, simple lists can specify every possible model of a given ring. A configuration file is just such a list, but the first item specifies the ring itself. A syntax diagram describes the items in the list. See Fig. 4.

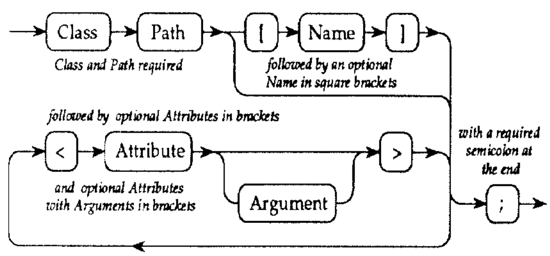

Fig. 4: Syntax Diagram for items in a Configuration File

The "Class" entry indicates what kind of model element to install next, such as: Ring, Injector, Map, Cavity, or others. The user can write and use additional classes.

The "Path" entry is a UNIX path to a descriptor file appropriate to the "Class" entry. Any path permissible by the c-shell (csh) is permissible as "Path" entry. User code can send messages to software modules by the "Name" entry, and receive results from them, during operation.

Optionally, the configuration file can ascribe attributes to a module, with arguments as required. Attributes permit the user to tailor the model by editing the configuration files. The descriptor files can remain a fairly small and constant set, and the configuration files can handle most of the volatility. Descriptor files take either of two simple forms, also described by syntax diagrams. Both forms resort to engineering units wherever applicable. Syntax diagrams provide a clear description of what is required of the user, and permit automatic generation of the parsing code [4].

IMPLEMENTATION

SimRing is written in Objective-C, an object-oriented language based on C. The Graphical User Interface uses a library called objcX [5]. This library emulates the NeXTSTEP class library for UNIX X/X-Window platforms. User-written code modules and user-written controller code could resort to Objective-C++, seamlessly. In this language, Objective-C code can send messages to objects and classes written in C++. Communication among modules in the ring, and between the Engine and Controller, resort to the server/proxy paradigm of distributed objects. The goal is to allow the user to run the Controller (GUI or code) on a local workstation, while running the Engine on a remote compute-server.

CONCLUSIONS

We have implemented a computer model for the rigid bunch motion of coherent bunch instabilities, based on power series maps to model the lattice and wake potentials.to model the wake fields. Numerical studies have been checked successfully against analytical calculations where these overlap. System analysis of related feedback systems by linear control theory has led to a deeper understanding of the dynamics as well as guidance for possible performance improvments. Object oriented programming and the server/proxy paradigm of distributed objects permit a modular computer implementation. Further work will focus on performance studies of the feedback systems with saturation effects and nonlinear dynamics.

ACKNOWLEDGMENTS

The authors would like acknowledge the following people for continuous support and encouragement during this work: S. Chattopadhyay, P. Kunz and M. Zisman.

REFERENCES

[1] M. Meddahi and J. Bengtsson, "Study of Transverse Coupled Bunch Instabilities by Using Non-Linear Taylor Maps for the Advanced Light Source (ALS)" proceedings of 1993 Particle Accelerator Conference, May 17-20, 1993, Washington, D. C.

[2] J. Bengtsson, D. Briggs and G. Portmann, "A Linear Control Theory Analysis of Transverse Coherent Bunch Instabilities Feedback Systems (The Control Theory Approach to Hill's Equation)" CBP Tech Note-026, PEP-II AP Note 28-93

[3] D. Briggs, et al., to be publ.

[4] M. E. Lesk, "Lex — A Lexical Analyzer Generator," Comp. Sci. Tech. Rep. No. 39. Bell Laboratories, October 1975.

[5] P. Kunz, et al., private comm. (objcX is available via anonymous FTP from ftp.slac.stanford.edu as file "/pub/sources/objcX-0.8.tar.gz".