

Methods for Detailed Modeling of the Longitudinal Phasespace Dynamics for Beams in Cyclic Accelerators

S. Stahl and J.A. MacLachlan
Fermilab*, P.O. Box 500, Batavia, IL 60510

Introduction

In this article the computer program ESME is presented as a tool for modeling the evolution of a distribution of particles in longitudinal phase space. General features of the code are discussed, and a specific example illustrating the use of the code is given.

The Program

The program was written primarily in 1981-82 to model specialized beam manipulations being developed for the Fermilab Antiproton Source. Its principal use was as a design tool, not as a facility for the detailed modeling of complex systems. In 1986 it was extended to examine transition crossing strategies in the Fermilab Booster. Among the improvements made at this time were the coupling of the single particle motion to the beam current through space charge, broad band wall impedance, and impedance of high-Q resonators.[1] The facility for phase feedback was made more realistic by allowing for a transfer function represented by a weighted average of the phase errors over many turns. This version has been useful in a wide variety of processes in several different accelerators and storage rings.[2] Recently interest has turned to the coupling of the motion of bunches to each other through these coupling impedances. Where nearby bunches are coupled through a resonant impedance the simple impedance formula is only a rough approximation of the coupling. A transient in the resonator from each bunch persists and evolves during the passage of many others. The transients produced by the later bunches reflect their response to transients produced by the earlier. The program has been extended to include detailed dynamics of this type, and there are some results for coupled bunch instability in the Fermilab Booster.[3] Although concerted effort has been made to keep the code as easy to use as the original, even where multi-particle dynamics are irrelevant new options and generality have resulted in more choices to make in setting up the input data. The process of choosing has been made as painless as possible by carefully developing reasonable defaults for all but the most fundamental parameters.

A side-effect of the most recent efforts at modification to the code has been the introduction of a VAX-based code management scheme which allows changes to be made and incorporated into the code with a minimum of effort. Machine dependencies, which have been kept to a minimum, are flagged for selection by a pre-processor.[5] This approach has allowed the code, though VAX-based, to be maintained on three different architectures; a VAX 8650, an Amdahl 5890, and a Floating Point Systems FPS-164.

The "core" of ESME consists of a routine which iterates the following pair of difference equations[2] on a turn-by-turn basis:

$$\Theta_{i,n} = (\Theta_{i,n-1} + 2\pi(1 - \frac{\Omega_{i,n}}{\Omega_{s,n}})) \quad (1)$$

and

$$E_{i,n} = E_{i,n-1} + \sum_{j=1}^n V_j \sin(h_j \frac{\Omega_{s,n}}{\Omega_{i,n}} \Theta_{i,n} - \phi_{s,n,j}) \quad (2)$$

where Ω is the revolution frequency, Θ and E are the azimuth and energy, and V the voltage. The subscript i is the particle index and n is the turn index. The subscript s implies that the quantity is at its synchronous value. The sum over j is one of the more recently implemented features of ESME, allowing one to track with multiple voltage sources. The ratio $\Omega_{s,n}/\Omega_{i,n}$ incorporates the kinematic non-linearity exactly and the lattice nonlinearity to whatever order in $\Delta p/p$ that the momentum compaction factor is known. In certain situations, the time required to iterate the difference equations once per turn becomes prohibitive. Consequently, ESME allows the difference equations to be iterated less often than once per turn. A recently implemented feature in ESME allows the difference equations to be implemented multiple times per turn (a feature which modifies the difference equations presented above slightly.)

The control of the voltages and phases can be rather involved when using multiple sources. ESME has options to allow the phases to be set explicitly by the user, to "synchronize" the phases to a programmed \dot{B} , to run the phases at a radial offset from the reference orbit, to operate one source as a Landau cavity *etc.* The voltage variation can be sigmoid, linear, isoadiabatic or according to a user-supplied interpolating polynomial. The phase feedback feature in the program allows for the implementation of a specific transfer function through the use of a user-supplied weighting function multiplying the phase error over time.

A number of options are provided to populate phase space with an initial distribution of particles, facilitating the investigation of a given set of rf parameters. Various combinations of matched, Gaussian, uniform random, *etc.* distributions are possible, as well as less realistic, but often useful examples such as a uniform grid, outline, and Gaussian raster. As referred to earlier, the entire distribution may be assigned a charge (distributed equally among the "macroparticles") and collective effects simulated. These effects may be computed and applied as often as one cares to between iterations of the difference equations.

The program is "driven" by FORTRAN namelist inputs activating specific routines. As mentioned earlier, an attempt has been made wherever possible to provide reasonable defaults for entries, reducing the amount of user input required. The variables describing the current "state" of the simulation (e.g., rf parameters, distribution coordinates) are held in common blocks. This state may be saved during the course of running the program, and the information used to drive a post-processor to produce graphical output or retained to resume the simulation later.

A variety of graphical output options are provided, including phase-space plots of the distribution and its projections, phase-space contours, the Fourier transform of the distribution, the voltage produced by the interaction of the beam with an impedance, as well as the voltage due to the response of a high-Q resonator to the beam current. A "history" of the rf parameters, distribution moments, user-selected Fourier amplitudes, a number of "spare" variables (for the user), and other interesting quantities can be retained throughout the simulation and plotted.

Finally, a more adventurous user may graft routines into ESME using one of the spare routines provided. These may be called on an *ad hoc* basis or from the main tracking routine. All variables in common are made available to these spare routines, including a number of spare

*Operated by the Universities Research Association, Inc. under contract with the U.S. Department of Energy.

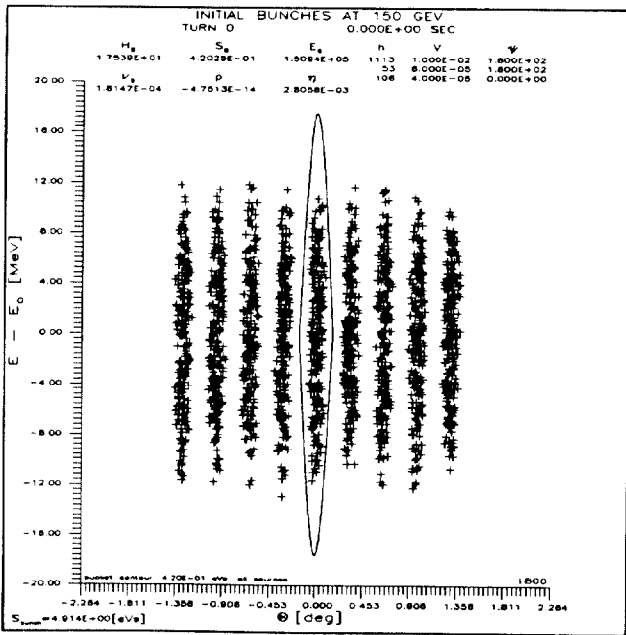


Figure 1: The initial distribution in the Main Ring, nine bunches at $h = 1113$ to be coalesced.

variables which are retained in the history and for which labels are provided. Many of the routines which have found their way into the code proper originated as such "spare" routines.

An Example

The code has been used in numerous applications, from modeling rf manipulations [4,6], to transition crossing with space charge [7], to coupled-bunch simulation [8]. Presented here is an example of debunching and bunch coalescing using the parameters of the Fermilab Main Ring.

Fig. 1 depicts nine bunches at 150 GeV in buckets produced by 10 kVolts at harmonic 1113. Each bunch is represented by 200 particles. Over the next 30000 turns, the voltage at harmonic 1113 is reduced to 800 Volts, and the bunches "debunch" into an $h = 53$ bucket at 80 Volts shaped by the presence of an $h = 106$ source at 40 Volts (Fig. 2). Then the lower frequency harmonic voltages are increased while the initial high-frequency harmonic is reduced, causing the bunch to rotate (Fig. 3). The program "traps" the bunch at the point at which it has rotated by 90 degrees, and then reintroduces harmonic 1113, giving us a much more tightly packed bucket (Fig. 4). An example of a history plot for this case is presented in Fig. 5. A mountain range plot depicting the entire process is shown in Fig. 6

References

- [1] J. A. MacLachlan, Fermilab FN-446(Feb. 1987), unpublished
- [2] J. A. MacLachlan, Fermilab FN-481(15 April 1988), unpublished and references therein.
- [3] A. Bogacz and S. Stahl, "Simulation of Coupled Bunch Mode Growth Driven by a High-Q Resonator: A Transient Response Approach", to appear in proc. of the Particle Accelerator Conf. held at Chicago, March 20-23, 1989.
- [4] J.E. Griffin, J.A. MacLachlan, Z.B. Qian, IEEE Trans. Nucl. Sci. 30#4 (1983), 2627-2629.

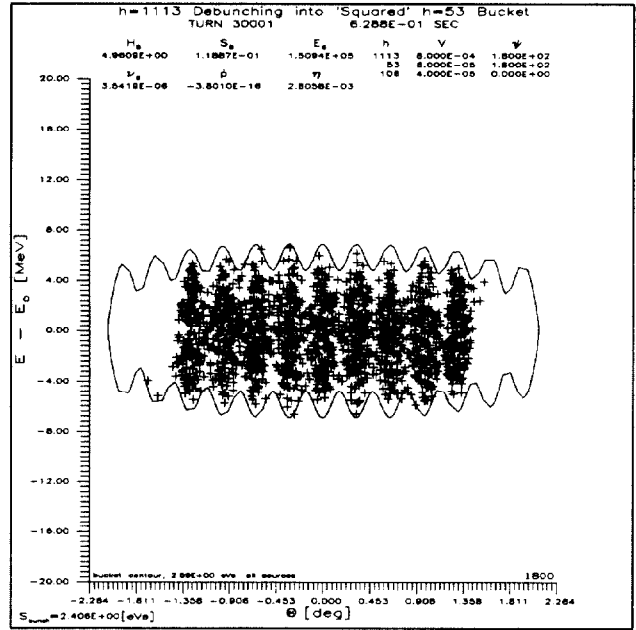


Figure 2: The distribution after having debunched for 30000 turns, low-frequency "squared" bucket contour.

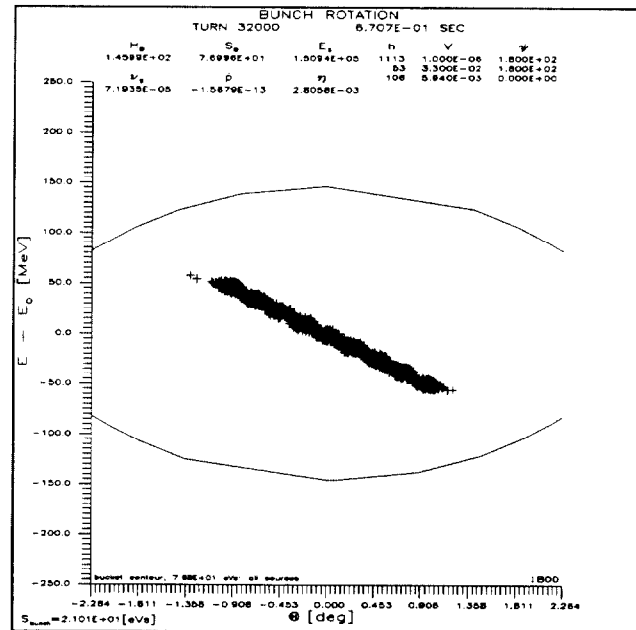


Figure 3: The rotating debunched distribution. Note that the vertical scale is approximately an order of magnitude larger than in the previous figures.

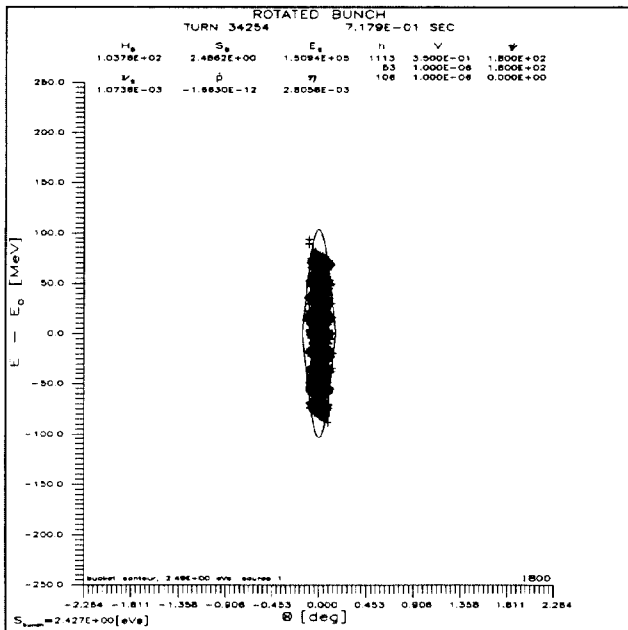


Figure 4: The distribution at the end of the rotation.

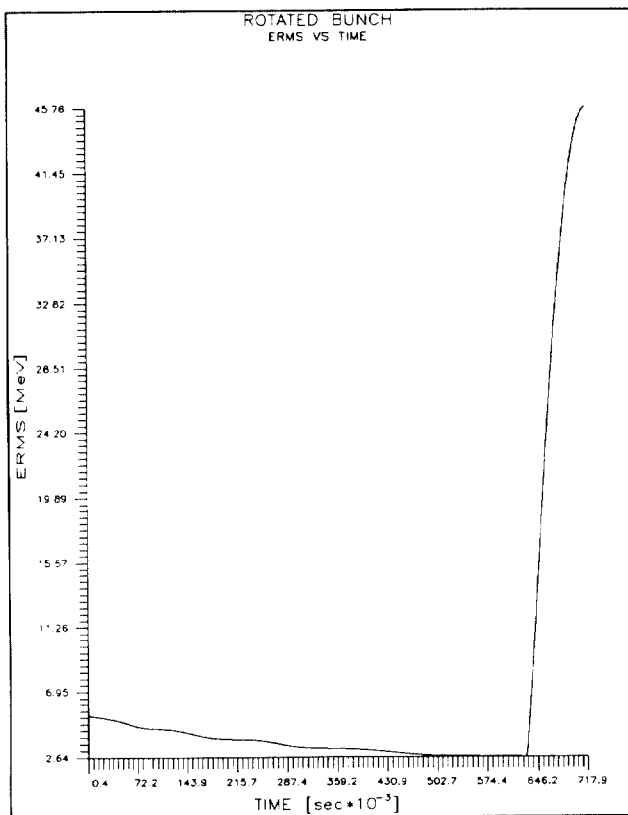


Figure 5: The rms width of the distribution in energy as a function of time.

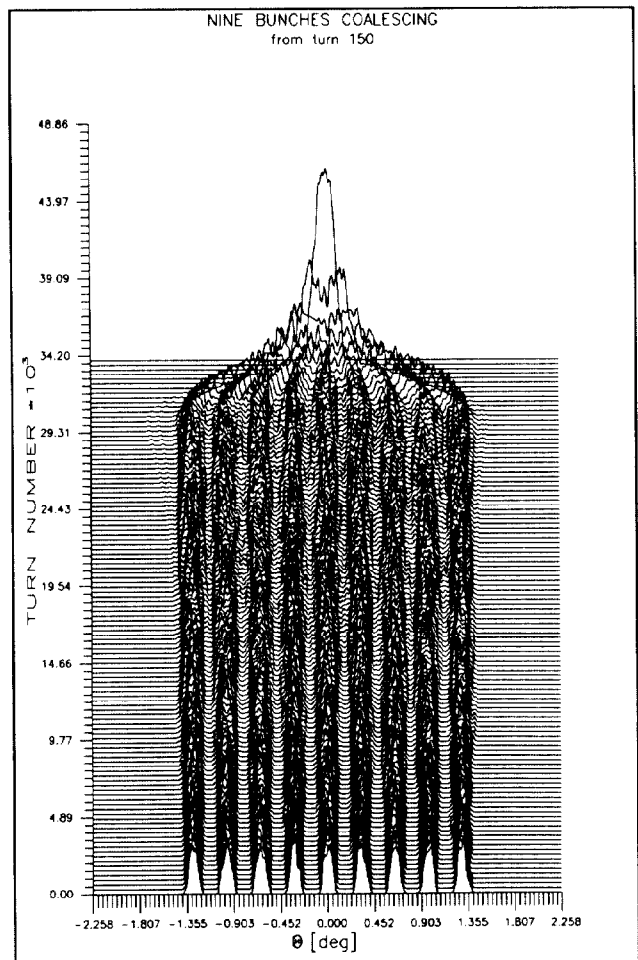


Figure 6: Mountain range plot depicting the evolution of the azimuthal profile of the bunches undergoing coalescing and rotation.

- [5] M.W. Eaton, "Expand and Xfort, Macro Expanders for CDF", Fermilab CDF-194(January 1986), unpublished.
- [6] S. Stahl and C. Ankenbrandt, IEEE PAC Proceedings, 1117 (1987).
- [7] P. Lucas and J.A. MacLachlan, IEEE PAC Proceedings, 1114 (1987).
- [8] S. Stahl and S.A. Bogacz, Phys. Rev. D **37**, 1300 (1988).