PEP-II/AP Note 95.39/LBL-37680/CBP Note-152

Comparisons of Beam-Beam Simulations*

Miguel Furman and Alexander Zholents

Center for Beam Physics
Accelerator and Fusion Research Division
Lawrence Berkeley National Laboratory, MS 71-259
University of California
Berkeley, CA 94720

and

Tong Chen

Accelerator Theory and Special Projects
Stanford Linear Accelerator Center, MS 26
Stanford, CA 94309

and

Dmitry Shatilov

Budker Institute of Nuclear Physics
630030 Novosibirsk 30
Russia

August 28, 1995

Abstract

We carry out a methodical comparison among the four beam-beam codes TRS, BBTRACK3D, LIFETRAC and TAIL under a restricted set of conditions for which such a comparison is meaningful. We first study the convergence rate of five slicing algorithms as the number of kicks goes to infinity and provide a criterion for the minimum number of kicks required for acceptable accuracy in a given situation. We then focus on turn-by-turn single particle tracking in 6-dimensional phase space in weak-strong mode for a thick-lens beam-beam interaction in the absence of damping and quantum excitation effects and lattice nonlinearities. When the codes make use of the the same thick-lens slicing algorithm, the results agree within computer accuracy. We also compute the tune shift with amplitude and compare the results with those from the first-order analytic calculation. The agreement is surprisingly good except when synchrotron sidebands are prominent. We then go on to include damping and quantum excitation and compute the 2-D particle distributions out to reasonably large amplitudes. The results, which we show in the form of contour level plots, agree within the statistical accuracy of the calculations. This article summarizes Ref. [1].

*Work supported by the Director, Office of Energy Research, Office of High Energy and Nuclear Physics, High Energy Division, of the US. Department of Energy under Contracts no. DE-AC03-76SF00098 and DE-AC03-76SF00515.

1 Introduction.

It is sometimes apparent that beam-beam simulations do not enjoy the same degree of respectability in today's accelerator physics community that other tools, such as single-particle tracking, do. One often hears, for example, that beam-beam simulations are better able to explain observed phenomena a posteriori rather than to predict them. Undoubtedly, the fundamental reason for this state of affairs is the complexity of the beam-beam interaction. A complete calculation would require the solution of Maxwell's and Newton's equations simultaneously for many billions of particles for millions of turns. It is clear that such a task is impossible with computers that are available today or that will be in the foreseeable future.

Nevertheless, much is known qualitatively and quantitatively about the beam-beam interaction in various regimes, and several codes have been developed that embody different approximations. Although the ultimate test of any beam-beam code is the correct and complete prediction of collider beam dynamics, it seems important for the time being to compare these codes with each other and with analytical results, and ensure that there is agreement whenever these comparisons are meaningful.

At its core, many of these codes have a common element: a thin-lens kick produced by a gaussian particle distribution. In this note we carry out a comparison among four beam-beam codes that involve this thin-lens kick. We start with the simplest case, namely the turn-by-turn tracking of a single particle colliding once per turn against an opposing gaussian bunch, and we compare the six-dimensional coordinates of the particle at every turn, in the absence of radiation damping, quantum excitation and lattice nonlinearities. We carry out the comparison for thin-lens and thick-lens beam-beam interaction, for five "slicing" algorithms, with or without synchrotron oscillations, for several initial conditions (but not in all possible combinations). When the codes do the same thing, the results indeed agree with each other within computer precision. We also compare the results for the calculated tune shift with amplitude with analytical results for the case of a thin lens. In the final step, we carry out a longer term simulation and produce the two-dimensional particle distribution in amplitude space with three of the codes. The agreement is very good, and the relatively small discrepancies are likely due to the difference in the algorithms used at this stage of the comparisons.

In all calculations presented here we use the "weak-strong" description of the beam-beam interaction. In this scheme the "strong" beam is passive and is represented by a gaussian lens (thin or thick) that is not altered by the other beam. The "weak" beam is dynamical, and we observe its behavior as a function of time as it collides repeatedly against the strong beam. In most, but not all, of the simulations carried out here, we use beam parameters that correspond closely to the PEP-II B factory [2], in which the electron beam plays the role of the strong beam and the positron that of the weak.

The four beam-beam codes we consider here are: TRS [3], LIFETRAC [4], TAIL [5] and BBTRACK3D [6]. The code TRS is a multiparticle strong-strong code that involves the soft-gaussian approximation. It is geared to assessing the luminosity performance of an e^+e^-collider. The codes LIFETRAC and TAIL are single-particle weak-strong codes geared to assessing the beam lifetime. The code BBTRACK3D is a single-particle weak-strong code geared to studying the dynamics of a single particle with specified initial conditions. Among its options, it allows different forms for the particle density of the strong beam, the gaussian being only one of them.

2 Slicing algorithms.

We assume that the longitudinal distribution of the opposing bunch is described by a gaussian density

$$\hat{\rho}_\ell(z) = \frac{e^{-z^2/2\sigma_z^2}}{\sqrt{2\pi}\sigma_z} \tag{1}$$

where the caret "ˆ" is meant to emphasize unit normalization. For the purposes of tracking simulations, we replace this density by a weighted superposition of N_s delta functions,

$$\hat{\rho}_\ell(z) \to \hat{\rho}_s(z) \equiv \sum_{k=-L}^{L} w_k \delta(z - z_k) \tag{2}$$

where $N_s \equiv 2L+1$ (we assume, as is customary, that N_s is an odd integer; if this is not the case, our calculation needs slight modifications). Each delta function gives rise to a kick at a location z_k weighted by w_k; these locations and weights must be determined according to a certain algorithm. The symmetry $\hat{\rho}_\ell(-z) = \hat{\rho}_\ell(z)$ implies that the kick locations and weights must obey the basic constraints $z_{-k} = -z_k$ and $w_{-k} = +w_k$. In addition, we require that the accumulated effects of the kicks should be the same as in the original distribution, *i.e.*, $\int dz \hat{\rho}_\ell(z) = \int dz \hat{\rho}_s(z) = 1$, which implies

$$\sum_{k=-L}^{L} w_k = 1 \tag{3}$$

For the thin-lens case ($N_s = 1$) there is a single kick at the center of the bunch with $z_0 = 0$ and $w_0 = 1$. For the thick lens case, on the other hand, there is, of course, an infinite number of possible algorithms to decide the weights and locations of the kicks consistent with the basic constraints. Here we examine only five possibilities. For the case $N_s = 5$ we list the kick locations and weights for all five algorithms in Table 1.

Algorithm #1 (equal spacing). In this case [7] the kicks are equally spaced and the weights are proportional to the gaussian density at z_k, namely

$$\frac{z_k}{\sigma_z} = \frac{2k}{N_s - 1}\left(1 + \frac{N_s - 3}{12}\right), \qquad w_k = \frac{\hat{\rho}_\ell(z_k)}{\sum\limits_{m=-L}^{L} \hat{\rho}_\ell(z_m)} \qquad (k = 0,\ \pm 1, \cdots,\ \pm L,\ N_s \geq 3) \tag{4}$$

Algorithm #2 (equal areas). In this case the gaussian distribution (1) is divided up into N_s "slices" of equal area (implying equal charge), and the kicks are located at the center of charge of each slice. The equality of the area of the slices implies that the weights are all equal, namely $w_k = 1/N_s$. Standard formulas for the area under a gaussian curve imply that the kick locations are given by

$$\frac{z_k}{\sigma_z} = \sqrt{2}\,\mathrm{erf}^{-1}\left(\frac{2k}{N_s}\right) \qquad (k = 0,\ \pm 1, \cdots,\ \pm L,\ N_s \geq 3) \tag{5}$$

Algorithm #3. This case [8, 9] is similar to the previous one, except that the kick locations are given by

$$\frac{z_k}{\sigma_z} = N_s\left[\hat{\rho}_\ell(l_k) - \hat{\rho}_\ell(l_{k+1})\right], \qquad k = 1, \cdots,\ L \tag{6}$$

where the l_k's are the edges of the slices. By arguments similar to those in the previous case, it is easy to see that, for $k > 0$,

$$\frac{l_k}{\sigma_z} = \sqrt{2}\,\mathrm{erf}^{-1}\left(\frac{2k-1}{N_s}\right) \qquad (k = 1,\ 2, \cdots,\ L+1,\ N_s \geq 3) \tag{7}$$

For $k < 0$, the l_k's are the mirrors of those for $k > 0$ (note that there is no $k = 0$ edge, and that the $k = \pm(L+1)$ edges are at $\pm\infty$). As in all cases, the central kick is at $z_0 = 0$, and the kicks for $k < 0$ are symmetrically located with respect to those for $k > 0$. The weights are the same as in the previous case, namely $w_k = 1/N_s$.

Algorithm #4. This is a modified combination of algorithms #1 and #3 in which

$$\frac{z_k}{\sigma_z} = \frac{1}{w_k}\left[\hat{\rho}_\ell(l_k) - \hat{\rho}_\ell(l_{k+1})\right], \qquad k = 1, \cdots,\ L \tag{8}$$

where the l_k's are the same as above, and where the locations for $k < 0$ are the mirror images of those for $k > 0$. The weights are proportional to $\sqrt{\hat{\rho}_\ell(z)}$, namely

$$w_k = \frac{\sqrt{\hat{\rho}_\ell(z_k)}}{\sum\limits_{m=-L}^{L} \sqrt{\hat{\rho}_\ell(z_m)}} \qquad (k = 0,\ \pm 1, \cdots,\ \pm L,\ N_s \geq 3) \tag{9}$$

In practice, the z_k's and w_k's are most easily found by iteration. It turns out that, of all five slicing algorithms described here, this algorithm #4 has the fastest rate of convergence as $N_s \to \infty$ (see the discussion below).

Algorithm #5. This algorithm consists of choosing the z_k's and w_k's in such a way that the area enclosed by the two functions $\int_0^z dz' \hat{\rho}_\ell(z')$ and $\int_0^z dz' \hat{\rho}_s(z')$ is minimal. This requirement leads to a set of nonlinear equations for the z_k's and w_k's which, as in algorithm #4, is most easily solved by iteration.

Table 1: Kick locations and weights for $N_s = 5$.

	Algorithm #1	Algorithm #2	Algorithm #3	Algorithm #4	Algorithm #5
	–1.166667	–1.281552	–1.399809	–1.59898	–1.44156
	–0.5833333	–0.5244005	–0.5319032	–0.67872	–0.63623
z_k's	0.0	0.0	0.0	0.0	0.0
	0.5833333	0.5244005	0.5319032	0.67872	0.63623
	1.166667	1.281552	1.399809	1.59898	1.44156
	0.1368561	0.2	0.2	0.17350	0.14943
	0.2280002	0.2	0.2	0.23222	0.22577
w_k's	0.2702873	0.2	0.2	0.26056	0.24960
	0.2280002	0.2	0.2	0.23222	0.22577
	0.1368561	0.2	0.2	0.17350	0.14943

Convergence rate of the five slicing algorithms. A reasonable requirement for any given algorithm is that the results should converge to a limit as $N_s \to \infty$. A reasonable requirement for all algorithms is that they should converge to the same answer in this limit. It remains an open problem to establish the optimal thick-lens slicing algorithm among the infinite number of possibilities. By "optimal algorithm" we mean that which yields, for a given finite number of kicks, the closest answer to the $N_s = \infty$ limit for a particular problem. This is clearly a very difficult problem: one cannot even set forth a universal criterion for such an optimization because such a criterion necessarily depends on many variables of the problem at hand, such as the working point, beam aspect ratios, etc.

In this section we try to solve a more modest problem: we study the convergence rate of the five slicing algorithms presented above as the number of kicks $N_s \to \infty$. Although this is clearly a more restricted problem than the one stated in the previous paragraph, the answer is still quite interesting because it shows clear systematics. As we show below, algorithm #4 emerges as the clear favorite among the five. Within this limited context, we also give an answer to the important practical question of how many kicks are enough for a given situation.

We proceed as follows: we first generate a weak bunch of 1,000 particles distributed gaussianly in the 6-D normalized phase space $(X_1, \ldots, X_6) \equiv (x/\sigma_{x+}, x'/\sigma_{x'+}, y/\sigma_{y+}, y'/\sigma_{y'+}, z/\sigma_{z+}, \Delta p/\sigma_{p+})$ where the subscript "+" is meant to emphasize that this is the weak (positron) beam. The coordinates are generated in an uncorrelated fashion, so that we may think of the distribution as that corresponding to a gaussian beam matched to the bare lattice at the IP. We then push this distribution once through the thick beam-beam lens produced by the strong beam for a given slicing algorithm and for a given number of kicks, and compare the resultant distribution with the one obtained by pushing the *same* initial distribution according to the "$N_s = \infty$" case which, for practical purposes, we define to be algorithm #4 with 300 kicks. Note that our criterion does not depend on any lattice parameter; it is designed to judge the beam-beam interaction by itself, as an isolated entity.

We compare the algorithms quantitatively by defining a dimensionless parameter Q for each algorithm as the sum of the rms deviations of the four transverse phase space coordinates of the final distribution from

those obtained from the $N_s = \infty$ case,

$$Q = \sum_{n=1}^{4} \sqrt{\langle (X_n - X_{n,\infty})^2 \rangle} \tag{10}$$

where $\langle \cdots \rangle$ is the average over the 1,000 particles. Obviously, the smaller is Q the better is the slicing algorithm. We present here only one case, corresponding to a flat beam with PEP-II-like parameters [2], listed in Table 2. Other cases, with qualitatively similar results, are described in Ref. [1].

Table 2: Parameters for the convergence rate tests.

β^*_{x+}/σ_{z+}, β^*_{y+}/σ_{z+}	37.5, 1.5
$\beta^*_{x-}/\beta^*_{x+}$, $\beta^*_{y-}/\beta^*_{y+}$	1.333, 1.333
σ_{x-}/σ_{x+}, σ_{y-}/σ_{y+}, σ_{z-}/σ_{z+}	1, 1, 1
σ_{x-}/σ_{y-}	25
ξ_{x+}, ξ_{y+}	0.03, 0.03

Figure 1 shows the results of plotting Q *vs.*the number of kicks N_s for the five algorithms. It is apparent that algorithm #4 has systematically the fastest convergence rate of the five. It is curious that algorithm #1 does not converge uniformly, although it becomes competitive with #4 for $\gtrsim 50$ or more kicks.

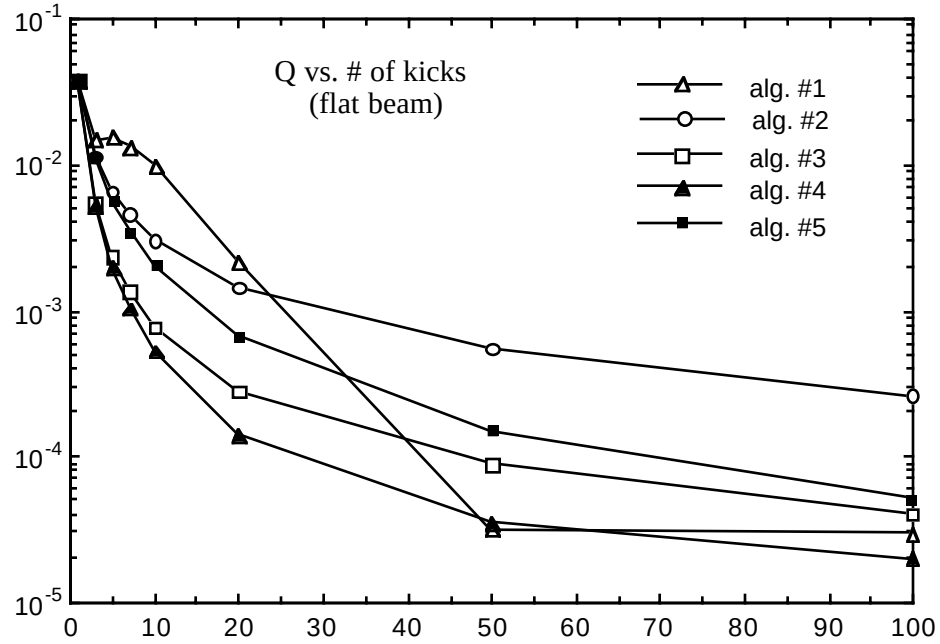

Figure 1: Convergence of the slicing algorithms: Q plotted vs. number of kicks N_s for flat beams (PEP-II-like parameters; see Table 2).

A criterion for the adequate number of kicks. It is important to decide how many kicks are enough for a given problem because, all other things being equal, the CPU time used in the calculation is proportional to N_s. We now provide a criterion for the minimum value required for N_s for issues pertaining to the beam

core (such as the beam blowup due to the beam-beam interaction, or, equivalently, the luminosity). The reasoning is as follows: due to the effects of radiation damping and quantum excitation, the rms beam size in the transverse dimension fluctuates by

$$\frac{\delta\sigma}{\sigma} \simeq \frac{1}{\sqrt{\tau}} \tag{11}$$

where τ is the damping time (in units of turn number) corresponding to the dimension considered. Because of these fluctuations, it is not justified to increase the accuracy of the beam-beam element beyond that corresponding to a value of Q given by

$$Q \simeq \frac{4}{\sqrt{\tau}} \tag{12}$$

(the factor 4 accounts for the four terms in Eq. (10)). Thus once we know the convergence rate of a given algorithm, the criterion is the following: the minimum value of N_s that gives the correct answer (within the statistical accuracy of the calculation) is that for which Q takes on the value given by Eq. (12).

For the case of PEP-II, with $\tau = 5400$, this yields $Q \simeq 0.05$. Therefore, as one can see from Fig. 1, any of the five algorithms gives the correct answer for $N_s = 3$. For other situations there are big differences in the number of slices required by each algorithm to converge to a result with a given accuracy [1]. Nevertheless, algorithm #4 has consistently the fastest convergence rate among the five possibilities shown here.

3 The thick lens beam-beam interaction.

In all cases described in this article we make the following assumptions: (a) The particles are relativistic so that we can neglect their self-interactions. (b) The bunch is sufficiently short that there are no nontrivial lattice elements at the IP in a region of size comparable to the bunch length. (c) There is no dispersion in this region around the IP. (d) There is no closed orbit distortion of the strong beam, intentional or accidental. (e) There are no phasing errors and no collective oscillations, longitudinal or transverse.

As the particles in the weak beam move through the strong beam, they encounter the N_s kicks in sequence. For each kick there are four steps (listed below) that describe the algorithm for the beam-beam kick experienced by a given particle in the weak beam. These four steps are repeated for each slice and must be carried out in the *actual sequence* of kicks encountered by the particle. When the weak beam consists of many particles, which is the generic case in the code TRS, one also has to repeat all these steps for all particles.

The four steps for a single kick are the following:

Step 1: Determine the collision point. As a consequence of the assumptions listed above, the bunch centers come together at the nominal (optical) IP. Thus we assume that they move towards each other according to $s_\pm = \pm ct$, so that time $t = 0$ corresponds to the instant of the central collision. If z_+ is the longitudinal position of the positron and z_- that of the electron slice (both measured relative to their respective bunch centers), then the azimuthal coordinates of the colliding positron (s_+) and opposing electron slice (s_-) at time t are

$$\text{positron: } s_+ = ct + z_+; \quad \text{electron slice: } s_- = -ct - z_- \tag{13}$$

(we take the convention that $z > 0$ means the head of the bunch regardless of its direction of motion). The collision point between the positron and the opposing slice is determined by setting $s_+ = s_- \equiv s_c$, which implies

$$\text{collision point: } s_c = \tfrac{1}{2}(z_+ - z_-) \tag{14}$$

Step 2: Determine the transverse coordinates. In all codes considered here the longitudinal as well as the transverse coordinates of the particles are *referred to the bunch center*. Since the beam-beam kick is represented by a thick lens whose strength varies during the course of the collision (due to the s dependence of the transverse size of the opposing bunch), we have to find the actual transverse coordinates

of the colliding particle. As a consequence of the assumptions listed above, the transformation from the bunch center to the actual collision point is a simple drift:

$$x \to x + s_c x', \qquad y \to y + s_c y' \tag{15}$$

while the slopes x' and y' remain unchanged.

Step 3: Actual kick. In this step the slopes of the particle are changed according to

$$x' \to x' + \Delta x'\left(x, y, \sigma_{x-}(s_c), \sigma_{y-}(s_c)\right), \qquad y' \to y' + \Delta y'\left(x, y, \sigma_{x-}(s_c), \sigma_{y-}(s_c)\right) \tag{16}$$

while x and y remain unchanged. In computing $\Delta x'$ and $\Delta y'$ we use x and y from Step 2 and the actual beam sizes of the opposing bunch at the collision point, given by

$$\sigma_{x-}(s_c) = \sigma_{x-}(0) \times \sqrt{1 + \left(s_c/\beta^*_{x-}\right)^2}, \qquad \sigma_{y-}(s_c) = \sigma_{y-}(0) \times \sqrt{1 + \left(s_c/\beta^*_{y-}\right)^2} \tag{17}$$

In all calculations discussed here we assume that the transverse distribution of the strong bunch is gaussian. Thus a particular slice of electrons centered at the origin and having horizontal and vertical rms sizes $(\sigma_{x-}, \sigma_{y-})$ is described by the transverse particle density

$$\rho_t(x, y) = \frac{\Delta N_-}{2\pi\sigma_{x-}\sigma_{y-}} \exp\left(-\frac{x^2}{2\sigma^2_{x-}} - \frac{y^2}{2\sigma^2_{y-}}\right) \tag{18}$$

which is normalized to the number of electrons ΔN_- contained in the slice (for the k-th slice, $\Delta N_- = w_k N_-$, where N_- is the total number of electrons in the bunch).

The electromagnetic kick $(\Delta x',\ \Delta y')$ received by a positron from a thin-slice electron bunch is written in concise form [10, 11] in terms of the complex error function $w(z)$[1] In tracking codes it is important to compute this function as fast as possible because this is the most CPU-intensive part of any beam-beam simulation that assumes a transverse gaussian shape for the bunches. In Ref. [1] we describe four methods of computing $w(z)$. They all yield results that are accurate to better than 1 part in 10^6. However, there are major differences in the computational speed of the different methods; we shall not address this issue here.

Step 4: Restore the coordinates to the reference point. This is the inverse of Step 2, namely

$$x \to x - s_c x', \qquad y \to y - s_c y' \tag{19}$$

while the slopes x' and y' remain unchanged.

Steps 2 and 4 do not cancel each other out because the slopes have changed in Step 3; therefore, in general, the beam-beam kick alters the coordinates as well as the slopes of the particles. For this reason, Step 4 is usually referred to as "disruption." The only circumstance in which there is no disruption occurs when *both* beams have zero bunch length. If only the strong beam has zero bunch length (*e.g.*, if it is represented by a thin lens), the weak beam will still suffer disruption due to the s_c dependence in Step 3 arising from the synchrotron oscillations of its particles.

4 Short-term single-particle tracking results.

Here we carry out turn-by-turn tracking for an individual particle at a time with given initial coordinates for a certain number N of turns. *For all cases in this section we completely neglect damping and quantum excitation* in order to eliminate numerical discrepancies arising from random number generators.

We assume that the lattice is decoupled and is described by a pair of 2×2 linear transport maps with given tunes. Similarly, we also assume that the longitudinal dynamics is described by a linear 2×2 map with a given synchrotron tune. We carry out a weak-strong calculation where the dynamical (or weak) beam is

[1]The function $w(z)$ is not to be confused with the weights w_k of the previous section, nor is the complex number z with the longitudinal coordinate z.

Table 3: Parameters used in the simulations.

	weak beam (e^+)	strong beam (e^-)
E [GeV]	3.1	9.0 †
β_x^* [m]	0.375	0.500
β_y^* [m]	0.015	0.020
σ_x [μm]	151.6 †	151.6
σ_y [μm]	6.063 †	6.063
ν_{0x}	0.57	0.57 †
ν_{0y}	0.64	0.64 †
σ_z [cm]	1.0	1.0
ν_s	0.0372	0.0523 †
σ_p/p	0.809×10^{-3}	0.615×10^{-3} †
N	5.630×10^{10} †	3.878×10^{10} †
ξ_{0x}	0.03	0.03 †
ξ_{0y}	0.03	0.03 †

† These parameters do not enter the weak-strong simulation, but we list them nevertheless for the sake of completeness.

the positron beam. The particle distribution of the electron beam is assumed to remain a three-dimensional gaussian whose transverse size is modulated by the beta functions, but whose emittances remain fixed at their nominal values.

The simulation data is obtained as follows: we first assign input values to all six normalized coordinates. The particle then undergoes a linear lattice transport, then a synchrotron map, and finally the beam-beam kick. Immediately after the beam-beam kick, its six normalized coordinates x/σ_x, $x'/\sigma_{x'}$, y/σ_y, $y'/\sigma_{y'}$, z/σ_z and $\Delta p/\sigma_p$ are recorded. This process is then repeated for N turns, with one line of data per turn. The tracking data is processed with the code PORTRAIT [12], from which we obtain all three phase space plots and their corresponding Fourier spectra. Each spectrum is produced as follows: we first compute

$$\tilde{X}_k = \sum_{n=1}^{N} X_n \, e^{2\pi i (n-1)k/N}, \qquad k = 0,\, 1,\, \cdots,\, N-1 \tag{20}$$

where X_n represents here either x/σ_x or y/σ_y or z/σ_z at turn n. We then normalize the spectrum by the largest of the absolute values $|\tilde{X}_k|$'s, and plot these normalized absolute values *vs.* k/N. We only plot the spectrum for the modes $k = 0,\, 1,\, \cdots,\, N/2$ on account of the well-known reflection symmetry of $|\tilde{X}_k|$ about $k = N/2$.

In addition to the spectrum, PORTRAIT computes all three dynamical tunes ν_x, ν_y and ν_z by numerically integrating the three phases over the N turns. We then form the linear combinations $n\nu_x + m\nu_y + l\nu_z$ where n, m and l are positive or negative integers[2] up to a certain maximum absolute value, and we plot a vertical dotted line whenever $n\nu_x + m\nu_y + l\nu_z$ (aliased to the interval $[0, 0.5]$) coincides with a local peak of the spectrum that is higher than a given threshold value. In this way we can identify resonances, which are labeled by the three integers n, m and l on the plots.

The simulation parameters are listed in Table 3. The values in this table are like those considered for the PEP-II B factory [2]; for the nominal PEP-II bunch collision frequency of 238 MHz, these parameters imply a nominal luminosity of $\mathcal{L}_0 = 3 \times 10^{33}$ cm^{-2} s^{-1}.

Comparison of the four codes. We now compare the results of the four codes considered here. In all cases we track for $N = 512$ turns, use $N_s = 5$ kicks, and use slicing algorithm #2. We compute $w(z)$

[2]Except that, without any loss of generality, we choose $n \geq 0$.

by simply invoking the IMSL® library function CERFE [13] except that the code TAIL uses the Pade approximant method [14]. The initial values of the coordinates are $x_0 = 3\sigma_x$, $y_0 = 1.5\sigma_y$, $z_0 = 3\sigma_z$ and $x'_0 = y'_0 = \Delta p_0 = 0$.

In Figs. 2 and 3 we show only the vertical phase space and spectrum obtained from each of the codes. We choose the vertical spectrum because, in our experience, it is more sensitive than the horizontal (and the longitudinal) in showing differences in the results. It can be seen that the codes agree with each other almost perfectly. The tiny differences (typically in the 4th or higher digit of the values of the coordinates) can be accounted for by the differences in the accuracy of the input values and of the different computers used for the tracking (codes TRS and BBTRACK3D were run on the same computer and their results do agree to computer accuracy).

We have carried out many more simulations with different initial conditions, different number of slices and different slicing algorithms. Ref. [1] contains but a small sample of these. The excellent agreement seen in Figs. 2 and 3 is typical of the larger set.

Algorithms for the complex error function. In Ref. [1] we also present a systematic comparison of four algorithms for the computation of $w(z)$: table interpolation to 3rd and 4th order, the Padé approximant method [14], and the IMSL library function CERFE [13]. The result of this exercise is that there are only small differences in accuracy (smaller than 1 part in 10^6), but large differences in computational speed. The table interpolation to 3rd order is the fastest of the four methods.

Effects of deliberate errors. In order to get an idea of the effects of nontrivial algorithmic errors, we show in Fig. 4 the results arising from incorrect coding. The first set in this Figure corresponds to an older version of TRS in which the kick from an individual slice was incorrectly modulated by the local beta function: Step 3 of the beam-beam kick (Eq. (16)) was coded as

$$x' \to x' + \left(\beta_{x-}(s_c)/\beta^*_{x-}\right)\Delta x', \qquad y' \to y' + \left(\beta_{y-}(s_c)/\beta^*_{y-}\right)\Delta y' \qquad \text{(incorrect!)} \tag{21}$$

Other than this difference, all tracking conditions were identical to those used in Figs. 2 and 3. One can see clear differences: the vertical amplitudes reach out to ~ 3 in Fig. 4 rather than to ~ 1.5 in Figs. 2 and 3, and the vertical spectra are substantially different.

As discussed in Sec. 3, Step 4 ("disruption") of the algorithm for the thick lens beam-beam interaction makes the sequence of kicks experienced by the tracked particle noncommutative. The second set of results in Fig. 4 shows the results of deliberately (and incorrectly) reversing the sequence of kicks experienced by the positron for the case $N_s = 5$. As in the previous example, there was no other difference in the tracking conditions from those used in Figs. 2 and 3. The phase spaces are not substantially different, but the vertical spectra are clearly different at the low end.

5 Comparison with analytic results.

In this section we compute the tune shift as a function of amplitude obtained from single-particle tracking with TRS and compare the results with first-order perturbation theory calculations. Again, we completely neglect radiation damping and quantum excitation and we use the thin-lens approximation, *i.e.* we take $N_s = 1$ slice in the beam-beam kick algorithm. Depending on the case studied, the particle may or may not perform synchrotron oscillations.

Case with no synchrotron motion. In this case the positron that is being tracked collides at the IP with a single-slice electron bunch. The rms beam sizes of the electron bunch at the IP σ_{x-} and σ_{y-} are understood to be evaluated at the IP.

The analytic calculation we use is described in Ref. [15], except for one detail which we will clarify below. The first step is to recall that the electromagnetic kick from one slice can be written as a two-dimensional vector as

$$(\Delta x', \Delta y') = -\frac{r_e\, N_-}{2\gamma_+}\left(\mathbf{E} + \mathbf{v}_+ \times \mathbf{B}\right) = -\frac{r_e\, N_-}{\gamma_+}\,\mathbf{E} \tag{22}$$

where the subscripts + and − refer to positrons and electrons, respectively. The electric field per unit charge produced by the electron beam, $\mathbf{E}(x, y, \sigma_{x-}, \sigma_{y-})$, is the solution of Poisson's equation, $\nabla \cdot \mathbf{E} = 4\pi\rho_t$, where $\rho_t(x, y, \sigma_{x-}, \sigma_{y-})$ is the transverse gaussian density, Eq. (18) (since the bunch is represented by a single slice, $\Delta N_- = N_-$). The fact that the magnetic and electric terms in the Lorentz force are equal is a consequence of the extreme relativistic approximation used (we also assume that the positron velocity $\mathbf{v}_+$ is antiparallel to the electron bunch velocity).

If we define the potential V as $\mathbf{E} = -\nabla V$, then the "phase averaged beam-beam parameter" (our nomenclature) of the kicked positron is given by [15]

$$\xi_i = -\frac{r_e\, N_-}{2\pi\gamma_+}\frac{\partial T_{00}}{\partial I_i}, \qquad i = x, y \tag{23}$$

where T_{00} is defined by

$$T_{00} = \int_0^{2\pi}\frac{d\theta_x}{2\pi}\int_0^{2\pi}\frac{d\theta_y}{2\pi}\, V\left(\sqrt{2\beta^*_{x+}I_x}\cos\theta_x, \sqrt{2\beta^*_{y+}I_y}\cos\theta_y\right) \tag{24a}$$

$$\equiv \left\langle V\left(\sqrt{2\beta^*_{x+}I_x}\cos\theta_x, \sqrt{2\beta^*_{y+}I_y}\cos\theta_y\right)\right\rangle \tag{24b}$$

Here the I's and θ's are the amplitudes and phases of the positron, respectively. The tune shifts of the positron $\Delta\nu_i$ are then obtained by solving the usual equations

$$\cos\left(2\pi(\nu_{0i} + \Delta\nu_i)\right) = \cos(2\pi\nu_{0i}) - 2\pi\xi_i \sin(2\pi\nu_{0i}), \qquad i = x, y \tag{25}$$

where the ν_{0i}'s are the "bare lattice" tunes.

In Ref. [15] the ξ's are assumed to be small enough that the approximation $\Delta\nu_i = \xi_i$ (which follows from Eq. (25) for small enough ξ) is valid. Hence in this approximation $\Delta\nu_i$ is given directly by Eq. (23). Our approach, which involves the extra step (25), can be thought of as a different perturbation expansion that reduces to the conventional one in the small-ξ limit, but that yields *the exact result when the perturbation force is linear.*[3]

It should be noted that $\sqrt{2\beta^*_{x+}I_x}$ and $\sqrt{2\beta^*_{y+}I_y}$ in Eq. (24) are nothing but the injection amplitudes x_0 and y_0 of the positron, respectively (we recall that in all our calculations the initial slopes vanish, $x'_0 = y'_0 = 0$). Therefore, by using the chain rule $\partial/\partial I_y = (\beta^*_{y+}/y_0)\partial/\partial y_0$ and $\mathbf{E} = -\nabla V$ we obtain

$$\xi_y = -\frac{r_e\, N_-\beta^*_{y+}}{2\pi\gamma_+\, y_0}\frac{\partial}{\partial y_0}\left\langle V\left(x_0\cos\theta_x, y_0\cos\theta_y\right)\right\rangle \tag{26a}$$

$$= \left\langle 2\cos^2\theta_y\, \xi_{y\ell}\left(x_0\cos\theta_x, y_0\cos\theta_y\right)\right\rangle \tag{26b}$$

and similarly for ξ_x. We have defined the "local vertical beam-beam parameter" $\xi_{y\ell}(x, y)$ as

$$\Delta y'(x, y) \equiv -4\pi\xi_{y\ell}(x, y)\frac{y}{\beta^*_{y+}} \tag{27}$$

where $\Delta y'(x, y)$ is given by Eq. (22).

As an example, we have used the PEP-II–like parameters listed in Table 3. As before, we tracked the particle with TRS for $N = 512$ turns, used one kick ($N_s = 1$), and used the IMSL® library [13] to compute $w(z)$. Radiation damping and quantum excitation were wholly neglected. The tune was computed with PORTRAIT. The tracked particle was injected with $x_0 = z_0 = 0$, and the vertical amplitude was varied in the range $0 \le y_0 \le 10\sigma_y$. For the analytic calculation, we integrated numerically Eq. (26b). Results are shown in Fig. 5. The agreement is almost perfect.

[3] One of us (MF) is indebted to Étienne Forest for a discussion on this point.

In Ref. [1] we also carried out this exercise for a round beam. In this case the phase integrals can be carried out and the result expressed in closed form. The agreement between tracking and the analytic result is also excellent.

Case with synchrotron motion. In case when the positron is performing synchrotron oscillations, it collides against the opposing thin-slice electron bunch at a longitudinal coordinate $s_c = z/2$ (see Eq. (14)) where z is the positron's longitudinal coordinate at the time of the collision. Thus the vertical kick it receives at the collision point (x, y, s_c) is given by

$$\Delta y'(x, y, s_c) = -\frac{r_e\, N_-}{\gamma_+}\; E_y\left(x, y, \sigma_{x-}(s_c), \sigma_{y-}(s_c)\right) \tag{28}$$

and similarly for $\Delta x'$. Here E_y is computed by using the actual beam sizes $\sigma_{x-}(s_c)$ and $\sigma_{y-}(s_c)$ of the electron bunch at the collision point, given by Eq. (17). Therefore it seems clear that the generalization to the present case of the phased-averaged vertical beam-beam parameter is

$$\xi_y = \left\langle 2\cos^2\theta_y\, \xi_{y\ell}\left(x_0\cos\theta_x, y_0\cos\theta_y, (z_0/2)\cos\theta_z\right)\right\rangle \tag{29}$$

where the local vertical beam-beam parameter $\xi_{y\ell}$ is defined by

$$\Delta y'(x, y, s_c) \equiv -4\pi\xi_{y\ell}(x, y, s_c)\frac{y}{\beta_{y+}(s_c)} \tag{30}$$

with a similar expression for the horizontal counterpart. Note that the phase average $\langle\cdots\rangle$ is now three dimensional,

$$\langle\cdots\rangle = \int_0^{2\pi}\frac{d\theta_x}{2\pi}\int_0^{2\pi}\frac{d\theta_y}{2\pi}\int_0^{2\pi}\frac{d\theta_z}{2\pi}\;(\cdots) \tag{31}$$

and that the beta function in Eq. (30) has the appropriate s_c-dependence.

Case when the positron is close to the axis. Fig. 5 shows the tracking results and analytic calculation of the vertical tune plotted as a function of the longitudinal launching amplitude z_0 of the positron (z_0 is normalized to the rms bunch length of the opposing bunch). The particle is tracked with TRS for $N = 512$ turns, with $N_s = 1$, using the IMSL® library calculation of $w(z)$ [13]. The particle is launched close to the beam axis ($x_0 = 0,\ y_0 = \sigma_y/10$), with $x_0' = y_0' = \Delta p_0 = 0$, and z_0 is varied in the range $0 \le z_0/\sigma_z \le 20$.

The three cases displayed in Fig. 5 correspond to different ratios of the beta functions of the two beams (in all 3 cases, however, the beams have the same aspect ratio, $\sigma_x/\sigma_y = 25$). These are dubbed "symmetric," "nominal," and "high asymmetry." The nominal case is exactly the same as that displayed in Table 3. The number of particles per bunch in all three cases are adjusted so that the nominal beam-beam parameters remain fixed at 0.03. Table 4 lists the relevant parameters.

In the three cases the beta functions at the IP of the positron beam are held fixed, and so is the beta-function ratio for each beam, namely $\beta_x^*/\beta_y^* = 25$. Similarly, the beam aspect ratio at the IP is fixed: $\sigma_x/\sigma_y = 25$ for all cases. What changes from one case to the next is the ratio of the beta function of one beam relative to the other: the ratio β_-^*/β_+^* takes on the values 1, 1.333... and 2.666... for the symmetric, nominal and high asymmetry cases, respectively. If the positron did not perform synchrotron oscillations, it is a priori obvious that the tune shift would be the same in all three cases. However, the fact that the beta functions of the *electron beam* are different makes the modulation of the *positron* beam-beam parameter vary from case to case due to the differences in the hourglass effect [16]. In fact, for a flat beam, a simple analytic calculation for the vertical beam-beam parameter of a positron oscillating longitudinally with maximum amplitude z_0 and with $x_0 \simeq y_0 \simeq 0$ shows [16] that $\xi_{y+}(z_0)$ scales like $\xi_{y+}(z_0) \sim \beta_{y+}(z_0/2)/\sqrt{\beta_{y-}(z_0/2)}$. This scaling shows that $\xi_{y+}(z_0)$ grows linearly when $z_0 \gtrsim \beta_{y-}^*$. Also, if β_{y+}^* is kept fixed, as we do in Table 4, the tune shift is larger the larger is β_{y-}^*. Actually, if this scaling formula (properly normalized) is inserted into Eq. (25), the resultant vertical tune is in good qualitative agreement with the more accurate calculations shown in Fig. 5.

Table 4: Parameters used in the three cases with synchrotron motion.

	symmetric		nominal		high asymmetry	
	e^+	e^-	e^+	e^-	e^+	e^-
E [GeV]	3.1	9.0	3.1	9.0	3.1	9.0
β_x^* [m]	0.375	0.375	0.375	0.50	0.375	1.0
β_y^* [m]	0.015	0.015	0.015	0.02	0.015	0.04
σ_x [μm]	131.3	131.3	151.6	151.6	214.4	214.4
σ_y [μm]	5.251	5.251	6.063	6.063	8.574	8.574
ν_{0x}	0.57	0.57	0.57	0.57	0.57	0.57
ν_{0y}	0.64	0.64	0.64	0.64	0.64	0.64
σ_z [cm]	1.0	1.0	1.0	1.0	1.0	1.0
ν_s	0.0372	0.0523	0.0372	0.0523	0.0372	0.0523
σ_p/p [10^{-3}]	0.809	0.615	0.809	0.615	0.809	0.615
N [10^{10}]	5.630	1.939	5.630	2.586	5.630	5.171
ξ_{0x}	0.03	0.03	0.03	0.03	0.03	0.03
ξ_{0y}	0.03	0.03	0.03	0.03	0.03	0.03

For the high asymmetry case, Fig. 5 shows that the tune turns over at $z_0/\sigma_z \gtrsim 12$. This is an artifact of the aliasing inherent in the turn-counting method used to calculate the tune. Aside from this effect, the results are in excellent agreement with the analytic calculation.

Case when the positron is away from the axis. In Ref. [1] we have also computed the tune shift when both the transverse and longitudinal amplitudes are nonzero. Space constraints do not allow us to present the results here. The result is that, when $z_0 > 0$, the tracking and the analytic results agree very well only for small values of y_0. In fact, the tracking results for the tune as a function of y_0 do not follow a smooth curve. In contrast, the analytic curves do behave smoothly. An analysis with PORTRAIT, however, shows that the synchrotron sidebands $\nu_y \pm \nu_s$ are prominent in this region of the amplitude space. It is virtually certain that these sidebands are responsible for the lack of agreement between the tracking and the analytic calculations (the analytic calculation is insensitive to these kinds of resonances, hence the smooth curves).

6 Long-term tracking: amplitude distributions.

One-dimensional distributions. The one-dimensional horizontal density is defined by

$$\frac{1}{N_0}\frac{dN}{d\hat{A}_x^2}, \quad \text{with} \quad \hat{A}_x^2 \equiv \frac{x^2 + (\beta_x x' + \alpha_x x)^2}{\sigma_x^2} \tag{32}$$

with corresponding expressions for the vertical counterparts. Here N represents the number of particle-turns at normalized amplitude $\hat{A}$, N_0 is the total number of particle-turns accumulated in the tracking run, x and x' are the position and slope of the particle, α_x and β_x are the usual lattice functions of the weak beam and σ_x is its nominal rms beam size. All these quantities are referred to the observation point, which we chose to be the interaction point.

In this kind of simulation, damping and quantum excitation effects play important roles since the shape of the particle distribution is a consequence of a dynamical equilibrium between these effects and the beam-beam interaction. In Ref. [1] we present the calculation of this density for the five slicing algorithms and for different number of slices with TRS. The result of this exercise is consistent with the discussion in Sec. 2 regarding the convergence rate of the slicing algorithms, and no new information can be extracted within the accuracy of the calculation.

Two-dimensional distribution. We now compare the results from the codes TRS, LIFETRAC and TAIL for the 2-dimensional distribution in $(\hat{A}_x, \hat{A}_y)$ space. In this case, we follow the custom of dealing with the density

$$\frac{1}{N_0}\frac{dN}{d\hat{A}_x d\hat{A}_y} \tag{33}$$

which is normalized to unity,

$$\int_0^\infty d\hat{A}_x \int_0^\infty d\hat{A}_y \; \frac{1}{N_0}\frac{dN}{d\hat{A}_x d\hat{A}_y} = 1 \tag{34}$$

For any physical particle distribution, this density vanishes whenever $\hat{A}_x = 0$ or $\hat{A}_y = 0$ due to a zero of the volume element. For reference, the nominal gaussian distribution is

$$\frac{1}{N_0}\frac{dN}{d\hat{A}_x d\hat{A}_y} = \hat{A}_x \hat{A}_y \, e^{-(\hat{A}_x^2+\hat{A}_y^2)/2} \qquad \text{(nominal gaussian)} \tag{35}$$

For the simulation we choose a large beam-beam parameter of 0.08 in order to enhance the tails of the distribution. We achieve this value by scaling N by a factor of 8/3 relative to Table 3. The parameters that are different from Table 3, including the damping times, are displayed in Table 5.

Table 5: Parameters used in the simulations for the 2-D distributions.

	weak beam (e^+)	strong beam (e^-)
N	15.01×10^{10}	6.895×10^{10}
ξ_{0x}	0.08	0.08
ξ_{0y}	0.08	0.08
τ_x [turns]	5400	5014
τ_y [turns]	5400	5014
τ_z [turns]	2700	2507

In these 2-D simulations we use 5 kicks ($N_s = 5$) for all three codes. However, the other conditions are not exactly the same: (1) TAIL uses the Padé approximant method [14] to calculate the complex error function rather than the IMSL® library [13]; (2) TRS uses a slightly different algorithm for radiation damping and quantum excitation from the other codes; and (3) LIFETRAC uses slicing algorithm #5 as opposed to algorithm #2 used by the other two codes.

Figure 6 shows the contour plots for the resultant simulation from the three codes. The contour levels are as follows: the first level is at a height $1/\sqrt{e}$ below the peak, and successive contour levels are at a constant ratio e below each other. The agreement among the codes is quite good given the differences between them.

The code TRS does "brute force" tracking. For the particular simulation shown here, we used 1,024 particles and tracked them for 500,000 turns. Thus we accumulated a total number of particle-turns $N_0 = 0.512 \times 10^9$. The program took 818 CPU minutes to run on the Cray-2S/8128 at NERSC. On the other hand, LIFETRAC and TAIL use an "acceleration algorithm" [17] that optimizes the generation of the tails of the distribution. For the simulation shown in Fig. 6, the results from LIFETRAC correspond to an effective number $N_0 = 4 \times 10^{11}$ of particle-turns and took ~ 100 CPU minutes on a VAX-6610 computer. The results from TAIL correspond to an effective number $N_0 = 6.3 \times 10^{11}$ of particle-turns and took ~ 200 CPU minutes to run on an IBM RS6000/375 computer.

7 Conclusions.

We have described a methodical comparison of four beam-beam codes with each other and with analytic calculations. We have carried out single-particle and multiparticle tracking calculations and have computed

the tune shifts with amplitude and the particle density distribution. In general, the agreement is almost perfect when the comparison is meaningful, and the tiny differences can be traced to round-off errors. We have studied the thin and thick lens approximations for the beam-beam interaction in weak-strong mode, different slicing algorithms, and different ways of computing the complex error function. This article is a brief summary of Ref. [1]; more details and more cases studied can be found in there. The good agreement exhibited by the cases shown here and in Ref. [1] are typical of a much larger set that space constraints do not allow us to present.

A study of the convergence rate as the number of kicks $N_s \to \infty$ of the five slicing algorithms shows that #4 is the most efficient. Based on the damping time and on the curves for the convergence rates, we have provided a criterion for the minimum number of kicks that must be used in a given situation for a given algorithm. For the case of the PEP-II nominal design, the adequate number of kicks is 3 for algorithm #4, and 15 for #1.

We found excellent agreement between the tracking results for the tune as a function of amplitude and the corresponding analytical calculations. These calculations were done for one slice, with or without synchrotron motion, for round and for flat beams. We found a discrepancy only in the case when the amplitudes of betatron and synchrotron oscillations are *both* large. In this case, however, the synchrotron sidebands of the vertical tune are prominent. Since the analytic calculation does not take these sidebands into account, the discrepancy is not meaningful.

By using the fourier spectrum of the single-particle motion as a probe, we uncovered errors in earlier versions of the code TRS. It turns out, however, that these errors lead to only minor effects in the multiparticle simulations carried out for PEP-II [2]. However, they might have been more important in other situations. One of our main motivations in carrying out the detailed single-particle comparisons was to look for these errors. It is gratifying that the spectrum of the motion provides such a useful magnifying glass through which to look at the beam-beam interaction.

The two-dimensional particle distributions in amplitude obtained from the three codes TRS, TAIL and LIFETRAC are in good agreement. In this case we do not expect the agreement to be perfect because the codes use different algorithms for slicing, radiation damping and quantum excitation.

As discussed in detail in Ref. [1], there are significant differences in computational speed (but not in accuracy) in the above-mentioned algorithms for the complex error function. In Ref. [1] we also discuss the effects from different slicing algorithms and different number of slices in the computation of the one-dimensional particle distributions; the conclusions from this particular exercise, however, are subsumed by those reached from the other calculations described here.

In summary, we have exhibited results in good agreement obtained with four different beam-beam codes. Although these codes are optimized for different purposes, it is clear that their basic "engines" are doing the same thing. Since the results also are in excellent agreement with analytical calculations, we conclude that the codes are correct to the extent that they involve the same approximations used in these calculations.

8 Acknowledgments.

We are grateful to NERSC for supercomputer support.

References

[1] M. Furman, A. Zholents, T. Chen and D. Shatilov, "Comparisons of Beam-Beam Code Simulations," CBP Tech Note-59/PEP-II AP Note 95.04, July 13, 1995.

[2] "PEP-II: An Asymmetric B Factory - Conceptual Design Report," June 1993, LBL-PUB-5379/SLAC-418/CALT-68-1869/UCRL-ID-114055/ UC-IIRPA-93-01.

[3] J. L. Tennyson, undocumented code "TRS," 1989.

[4] D. Shatilov, "Beam-Beam Simulations at Large Amplitudes and Lifetime Determination," BINP 92-79, Novosibirsk (in Russian).

[5] T. Chen, J. Irwin and R. Siemann, "Simulation of the Beam Halo from the Beam-Beam Interaction," Phys. Rev. **E 49**, 2323 (1994); "Simulation of the Beam Halo from the Beam-Beam Interaction in LEP," SLAC-PUB-6432, Feb. 1994, to be published in the Proc. 4th annual LEP performance workshop, Chamonix, France, Jan. 1994; "Simulation of the Beam-Beam Lifetime for LEP," SLAC-PUB-6561, June 1994, to be published in the Proc. 4th EPAC, London, England, June 27-July 1st., 1994.

[6] M. A. Furman, "The Beam-Beam code BBTRACK3D" (to be documented).

[7] J. L. Tennyson, "Tune Considerations for APIARY 6.3D," ABC-28, August 1991.

[8] K. Hirata, H. Moshammer and F. Ruggiero, "A Symplectic Beam-Beam Interaction with Energy Change," Part. Accel. **40**, 205 (1993).

[9] S. Krishnagopal, "Beam-Beam Dynamics with Round Beam Profiles," Ph. D. dissertation, Cornell University, January 1991.

[10] M. Bassetti and G. A. Erskine, "Closed Expression for the Electric Field of a Two-Dimensional Gaussian Charge," CERN-ISR-TH/80-06.

[11] M. A. Furman, "Compact Complex Expressions for the Electric Field of 2-D Elliptical Charge Distributions," Am. J. Phys. **62** (12), Dec. 1994, 1134–1140.

[12] M. A. Furman, "The Tracking Analysis Program PORTRAIT" (to be documented).

[13] IMSL Math/Library® Reference Manual, v. 10.0.

[14] Y. Okamoto and R. Talman, "Rational Approximation of the Complex Error Function and the Electric Field of a Two-Dimensional Charge Distribution," CBN 80-13, Sept. 1980.

[15] R. Siemann,"The Beam-Beam Interaction in e^+e^- Storage Rings," Proc. Joint US-CERN School on Particle Accelerators: *Frontiers of Particle Beams, Factories with e^+e^- Rings*, Benalmádena, Spain, October 1992, pp. 327–363 (Springer Verlag Lecture Notes in Physics no. 425, M. Dienes, M. Month, B. Strasser and S. Turner, eds.)

[16] M. A. Furman, "The Hourglass Reduction Factor for Asymmetric Colliders," ABC-21/ESG Tech Note 161, April, 1991 (rev. Aug. 1991); "Hourglass Effects for Asymmetric Colliders," LBL-30833, Proc. 1991 Particle Accelerator Conf., San Francisco, May 6–9, 1991, p. 422.

[17] J. Irwin, "Simulation of Tail Distributions in Electron-Positron Circular Colliders," Proc. Third Advanced ICFA Beam Dynamics Workshop (Beam-Beam Effects in Circular Colliders), I. Koop and G. Tumaikin, eds., Novosibirsk, May 29–June 3, 1989, p. 123.

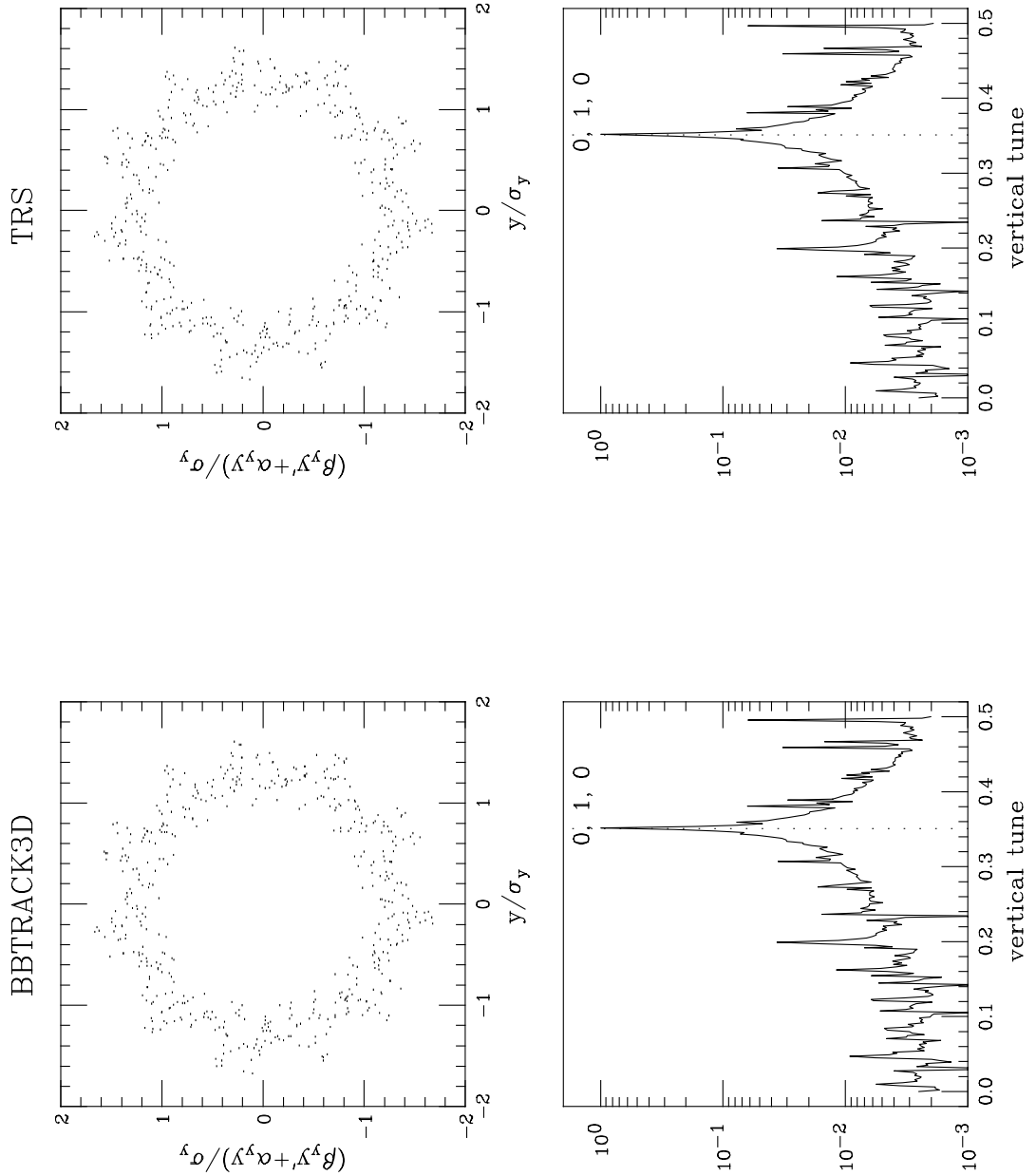

Figure 2: Single particle tracking (BBTRACK3D and TRS, $N = 512$, $N_s = 5$, slicing alg. #2).

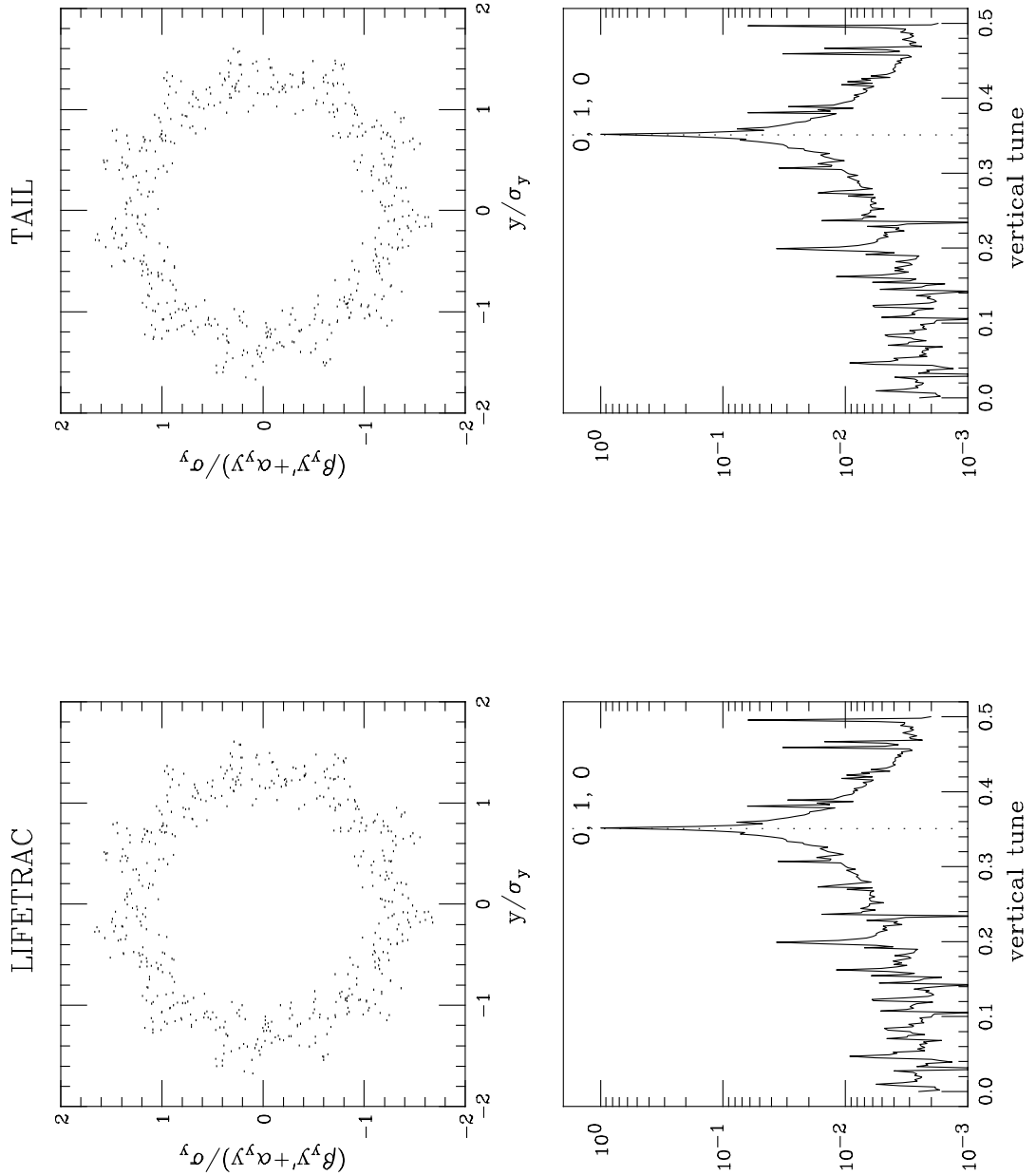

Figure 3: Single particle tracking (LIFETRAC and TAIL, $N = 512$, $N_s = 5$, slicing alg. #2).

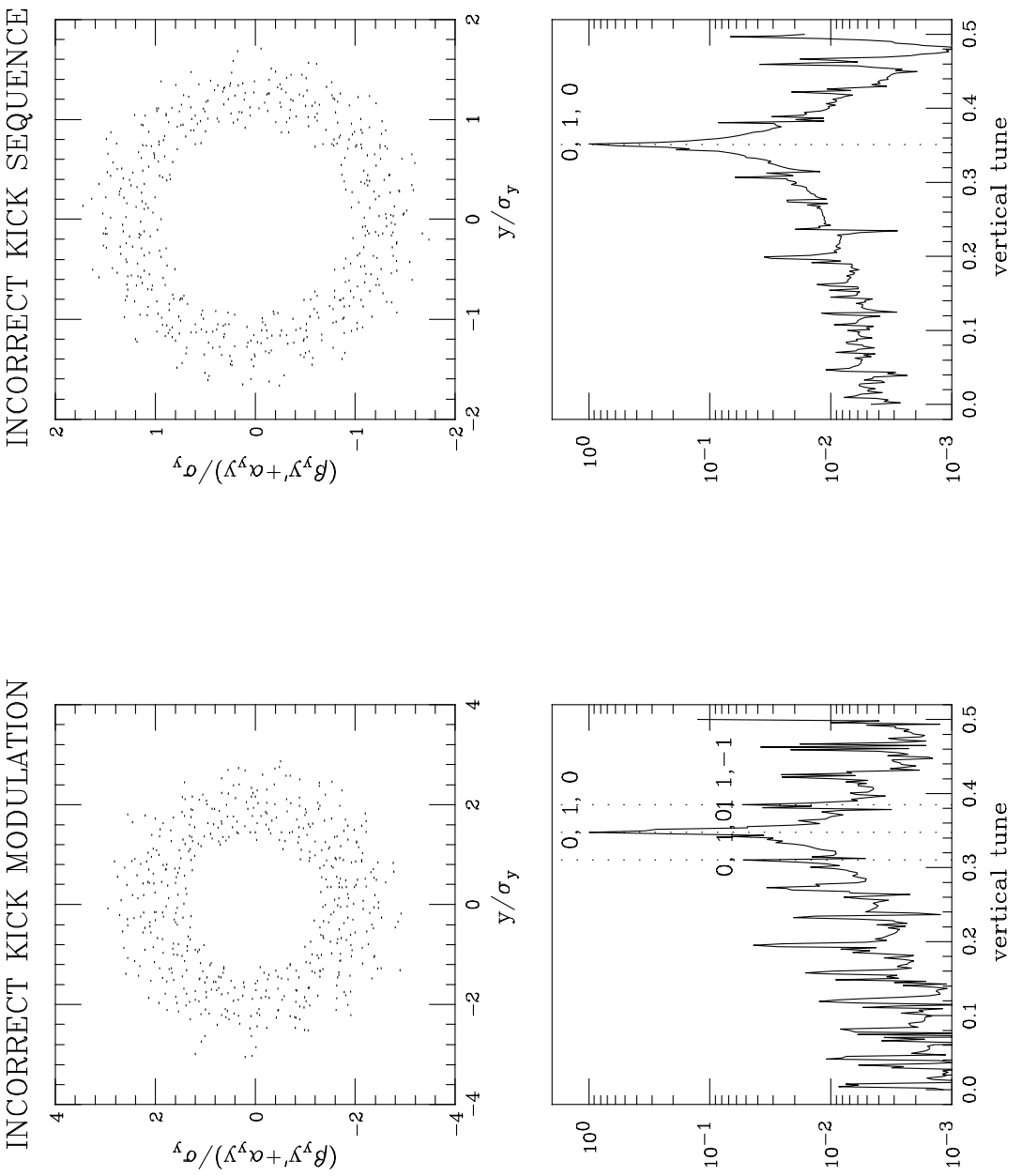

Figure 4: Single particle tracking with two deliberate errors: incorrect modulation of the kicks (Eq. (21), or inverted kick sequence (TRS, $N = 512$, $N_s = 5$, slicing alg. #2).

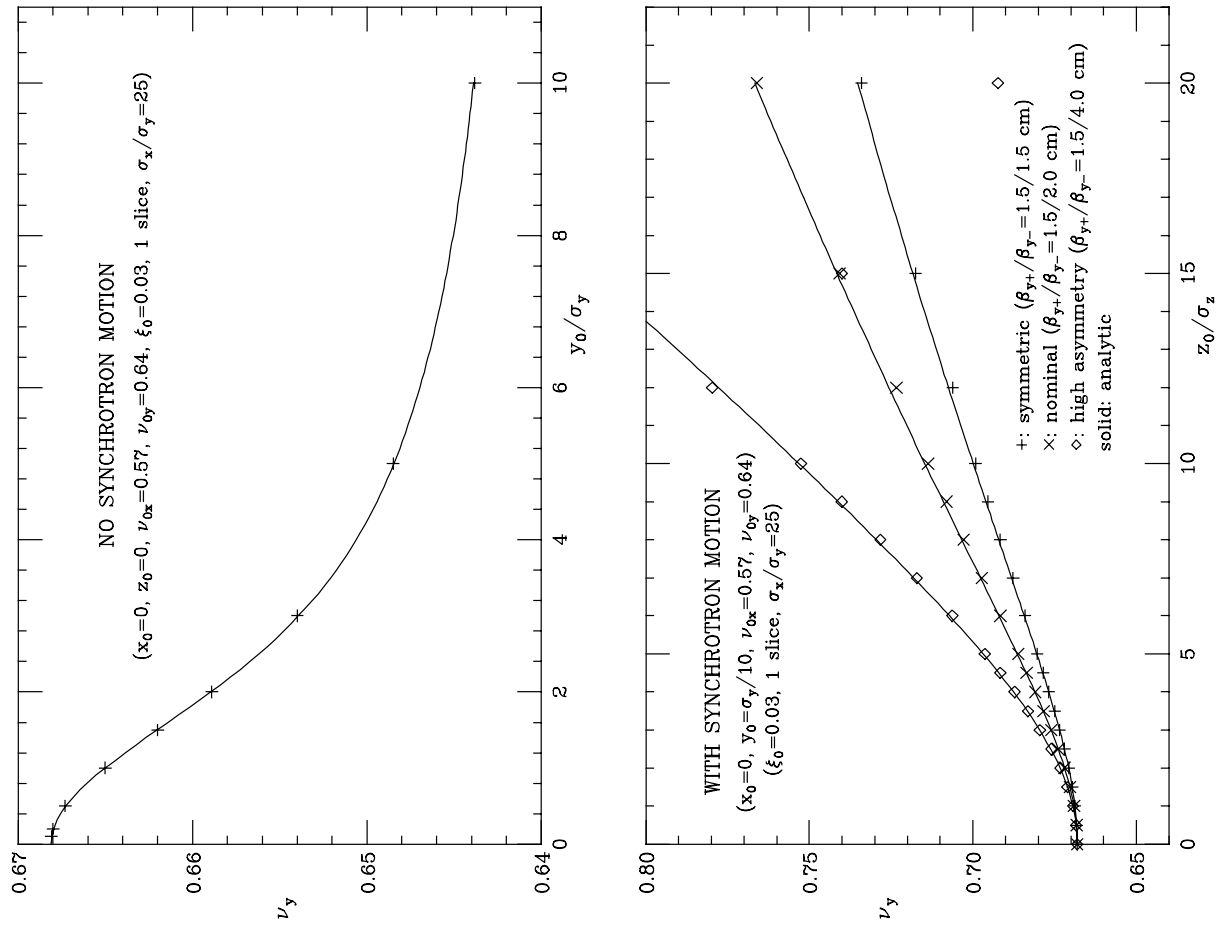

Figure 5: Top: vertical tune *vs.*vertical amplitude with no synchrotron motion; bottom: vertical tune *vs.*longitudinal amplitude for fixed (but small) vertical amplitude. Tracking results (crosses and diamonds) for a single positron colliding against a thin-lens electron beam were obtained with TRS with $N_s = 1$. In all cases the electron beam aspect ratio is $\sigma_x/\sigma_y = 25$; the differences between the three sets of data in the bottom figure are due to the hourglass effect.

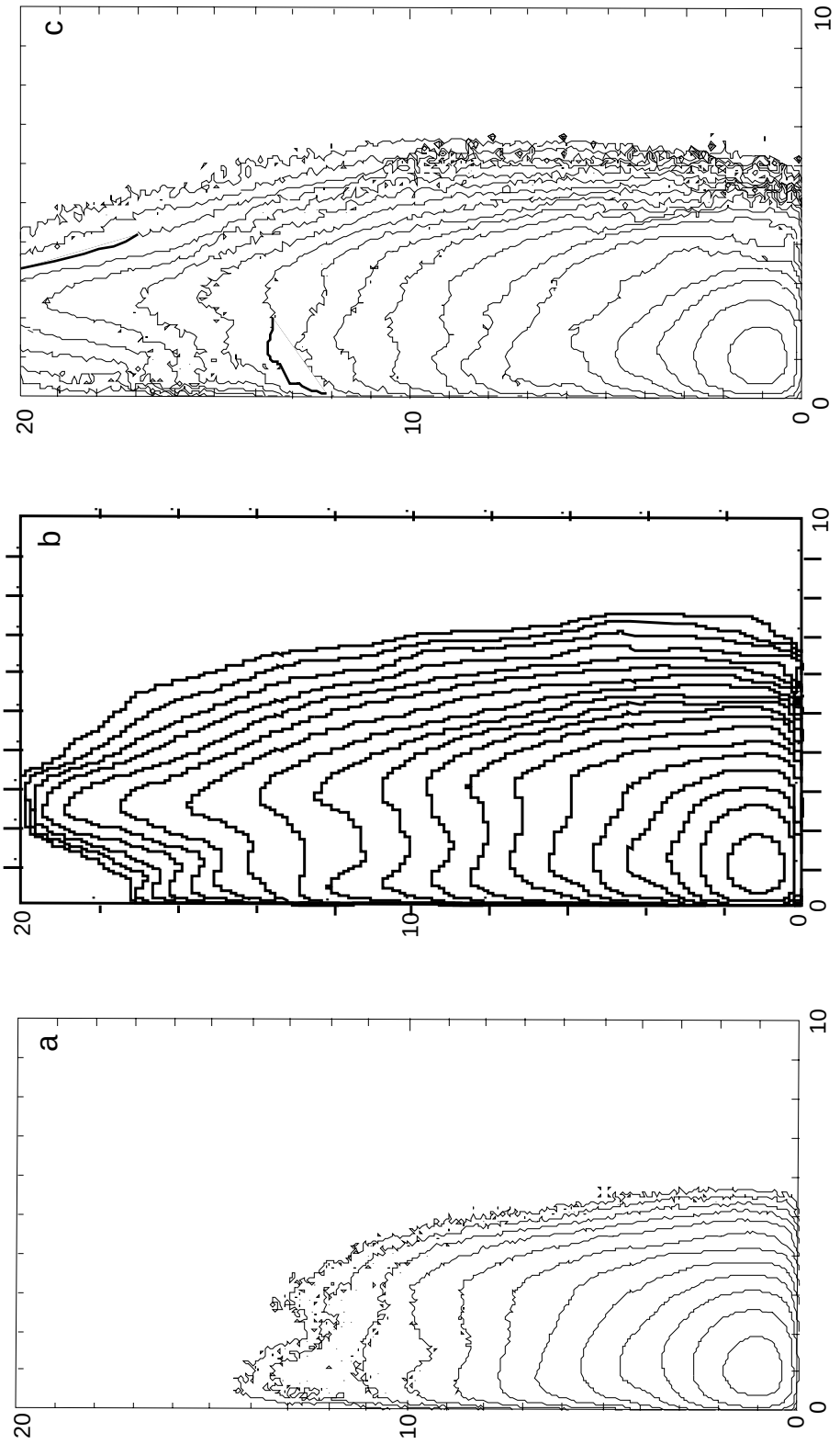

Figure 6: The 2-D amplitude distribution. (a): TRS; (b): LIFETRAC; (c): TAIL. All three cases use 5 kicks; other conditions are described in the text.