CBP Tech Note-59
PEP-II/AP Note 95.04

Comparisons of Beam-Beam Code 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

March 14, 1996

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 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 the first-order analytic calculation. The agreement is surprisingly good except in a few cases when synchrotron sidebands, which are clearly identified, affect the tracking results. Furthermore, we carry out, within TRS, a comparison for five different slicing algorithms and four algorithms for the computation of the complex error function (but not in all combinations). We then go on to include damping and quantum excitation and compute the 1-D and 2-D particle distributions out to reasonably large amplitudes. For the 1-D distributions, obtained with TRS, we compare the effects of different slicing algorithms and of different number of kicks up to 15. We compute the 2-D distributions with the codes TRS, LIFETRAC and TAIL. The results, which we show in the form of contour level plots, agree within the statistical accuracy of the calculations. Finally, we study the convergence rate of the 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.

Contents

List of Figures

List of Tables

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 and four algorithms for the computation of the complex error function, 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. The agreement is excellent except when synchrotron sidebands (which are not accounted for in the analytic approach) are significant. We then go on to include radiation damping and quantum excitation and compare the effects of using different slicing algorithms and different number of kicks on the one-dimensional particle distribution in amplitude space. This we do with only one of the codes considered here. 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. Finally, we carry out a methodical comparison of the convergence rate of the five slicing algorithms when the number of kicks goes to infinity. We conclude that one of the algorithms is the clear favorite among the five discussed. From the calculation of the convergence rate, we extract a criterion for the minimum number of kicks necessary for an accurate calculation in a given situation.

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 [1], 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 [2], LIFETRAC [3], TAIL [4] and BBTRACK3D [5]. 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 code has many options, including the possibility of performing single- or multi-particle tracking in weak-strong mode. The codes LIFETRAC and TAIL are single-particle weak-strong codes geared to assessing the beam lifetime. They involve similar (but not identical) "acceleration algorithms" by virtue of which most of the computer time is spent tracking the large-amplitude particles responsible for the tails of the beam rather than the core [6]. 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.

In Section 2 we describe the thick-lens beam-beam interaction in detail; the different ingredients are as follows: Sec. 2.1: slicing algorithms; Sec. 2.2: algorithms for the computation of the complex error function; Sec. 2.3: the beam-beam kick from one slice. In Section 3 we describe the transverse and longitudinal phase-space maps. In Section 4 we describe the map representation of the radiation damping and quantum excitation. Section 5 contains the short-term, single-particle, tracking results in the absence of radiation damping and quantum excitation: Sec. 5.1 compares the results, obtained with BBTRACK3D, from the 5 slicing algorithms; Sec. 5.2 compares the results from BBTRACK3D for the four algorithms for the complex error function; Sec. 5.3 compares the results from the four codes. In Section 6 we compare the single-particle tracking results obtained with TRS with analytic calculations for the tune shift as a function of amplitude: Sec. 6.1 contains the results for the case with no synchrotron motion (for round beams and for flat beams); Sec. 6.2 contains the results for the case with synchrotron motion (with and without transverse displacement). In Section 7 we collect the results for the one-dimensional particle distributions in amplitude obtained from multiparticle tracking with TRS. We accumulate $\sim 5\times 10^6$ particle-turns, which allows a reasonably accurate evaluation of the vertical distribution out to an amplitude $A_y \sim 5\sigma_y$. We compare the effects on the particle distribution from the different slicing algorithms and from using different number of kicks, up to 15. Section 8 contains the results from TRS, LIFETRAC and TAIL for the two-dimensional particle distributions in amplitude, which we present in the form of contour level plots. For the purposes of this exercise we choose a beam-beam parameter of 0.08; for this rather large value, the tails extend to fairly large vertical amplitude. TRS computes these distributions by "brute force" tracking, accumulating $\sim 0.5\times 10^9$ particle-turns; this allows a reasonable evaluation of the vertical distribution out to a vertical amplitude $A_y \sim 12\sigma_y$. LIFETRAC and TAIL, as mentioned above, use an acceleration algorithm that allows the evaluation of the particle distribution out to significantly larger amplitudes in much less time than the brute force method. There is good agreement among the three codes. In Section 9 we discuss the convergence rate of the five algorithms presented in Sec. 2.1 as the number of kicks is increased to large numbers, and present a criterion for the minimum number of kicks required for any given situation. Our conclusions and acknowledgments are collected in Sections 10 and 11, respectively.

2 The thick lens beam-beam interaction.

The thick-lens model of the beam-beam interaction consists of several components which we will describe in detail along with the assumptions involved in each step.

2.1 Slicing the opposing bunch longitudinally into thin slices.

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 "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

$$z_{-k} = -z_k \tag{3a}$$

$$w_{-k} = w_k \tag{3b}$$

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

$$\sum_{k=-L}^{L} w_k = 1 \tag{4}$$

For the thin-lens case ($N_s = 1$), there is a single kick at the center of the bunch, namely

$$z_0 = 0, \qquad w_0 = 1 \qquad\quad (N_s = 1), \tag{5}$$

but for the thick lens case there is, of course, an infinite number of possible algorithms to decide the weights and locations of the kicks consistent with Eqs. (3). 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. A discussion of the convergence rate as $N_s \to \infty$ of the five algorithms studied in this note is presented in Sec. 9. For $N_s = 5$, the z_k's and w_k's are listed in Table 1.

2.1.1 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

$$\left.\begin{aligned} \frac{z_k}{\sigma_z} &= \frac{2k}{N_s - 1}\left(1 + \frac{N_s - 3}{12}\right) \\ w_k &= \frac{e^{-z_k^2/2\sigma_z^2}}{\sum\limits_{m=-L}^{L} e^{-z_m^2/2\sigma_z^2}} \end{aligned}\right\} \qquad k = 0,\ \pm 1, \cdots,\ \pm L \qquad (N_s \geq 3) \tag{6}$$

2.1.2 Algorithm #2 (equal areas).

In this case the gaussian distribution (1) is divided up into N_s "slices" of equal area (implying equal charge), as shown in Fig. 1 for the case $N_s = 5$, whose edges we label l_k, with $k = \pm 1,\ \pm 2, \cdots,\ \pm(L+1)$. The equality of the area of the slices implies that the weights are all equal, namely $w_k = 1/N_s$. Since the kicks are located at the center of charge of each slice, one can easily see that

$$\int_0^{z_k} dz \hat{\rho}_\ell(z) = \frac{k}{N_s} \tag{7}$$

Now the area under a normalized gaussian is given by the formula

$$\int_0^{z} dz' \hat{\rho}_\ell(z') = \frac{1}{2}\mathrm{erf}\left(\frac{z}{\sqrt{2}\sigma_z}\right) \tag{8}$$

from which we find

$$\left.\begin{aligned} \frac{z_k}{\sigma_z} &= \sqrt{2}\,\mathrm{erf}^{-1}\left(\frac{2k}{N_s}\right) \\ w_k &= \frac{1}{N_s} \end{aligned}\right\} \qquad k = 0,\ \pm 1, \cdots,\ \pm L \qquad (N_s \geq 3). \tag{9}$$

2.1.3 Algorithm #3.

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

$$\frac{z_k}{\sigma_z} = N_s \left[\hat{\rho}_\ell(l_k) - \hat{\rho}_\ell(l_{k+1})\right], \qquad k = 1, \cdots, L \tag{10}$$

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 $k > 0$,

$$\frac{l_k}{\sigma_z} = \sqrt{2}\,\mathrm{erf}^{-1}\left(\frac{2k-1}{N_s}\right), \qquad k = 1,\ 2, \cdots,\ L+1 \qquad (N_s \geq 3) \tag{11}$$

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 $w_k = 1/N_s$.

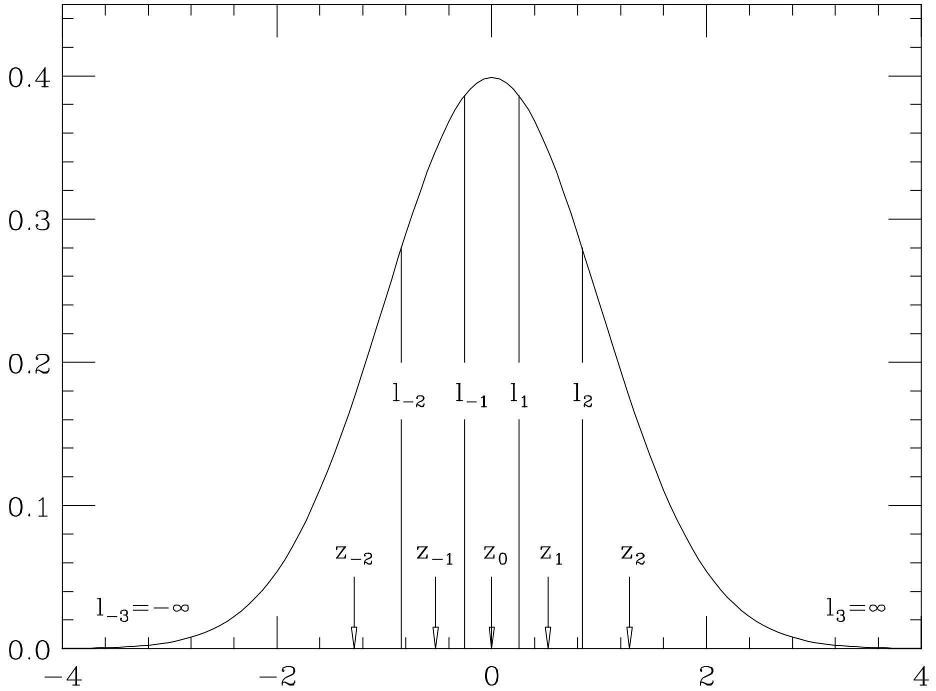

Figure 1: The gaussian density, Eq. (1) with $\sigma_z = 1$, is divided up into $N_s = 5$ equal-area slices whose edges are labeled l_k, according to Algorithm #2 (the outermost edges are at infinity). The center of charge of each slice is located at z_k. The l_k's and z_k's are at their actual locations obtained from Eqs. (9) and listed in Table 1.

2.1.4 Algorithm #4.

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

$$\frac{z_k}{\sigma_z} = \frac{1}{w_k} \left[\hat{\rho}_\ell(l_k) - \hat{\rho}_\ell(l_{k+1})\right], \qquad k = 1, \cdots, L \tag{12}$$

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

$$w_k = \frac{e^{-z_k^2/4\sigma_z^2}}{\sum\limits_{m=-L}^{L} e^{-z_m^2/4\sigma_z^2}}, \qquad k = 0,\ \pm 1, \cdots,\ \pm L \qquad (N_s \geq 3) \tag{13}$$

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 Sec. 9).

2.1.5 Algorithm #5.

This algorithm consists of adjusting 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. Fig. 2 shows these two functions for the case $N_s = 5$. As in algorithm #4, the z_k's and w_k's are most easily found by iteration.

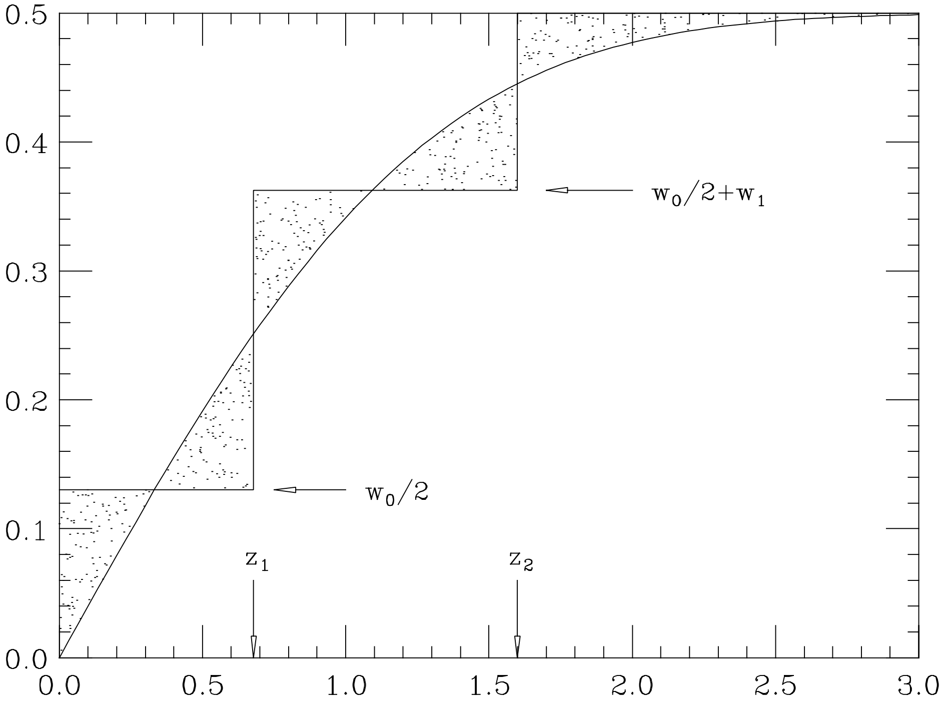

Figure 2: The functions $\int_0^z dz' \hat{\rho}_\ell(z')$ (smooth curve) and $\int_0^z dz' \hat{\rho}_s(z')$ (steps) for $N_s = 5$ plotted vs. z for $\sigma_z = 1$. In slicing algorithm #5, the z_k's and w_k's are found by minimizing the total area between the curves, shown shaded. The arrows show the actual values for the z_k's and w_k's obtained from this procedure (numerical values are listed in Table 1). Note that Eq. 4 implies that $w_0/2 + w_1 + w_2 = 1/2$.

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

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

2.2 Algorithms for the complex error function.

In the simulations reported in this note, we also assume that the transverse distribution of the strong bunch is gaussian. As mentioned above, the distribution is divided up into thin slices; 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

$$\rho_t(x,y) = \frac{\Delta N_-}{2\pi\sigma_{x-}\sigma_{y-}} \exp\left(-\frac{x^2}{2\sigma_{x-}^2} - \frac{y^2}{2\sigma_{y-}^2}\right) \tag{14}$$

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 received by a positron from this thin-slice electron bunch is succinctly written in complex form as [10, 11]

$$\Delta y' + i\Delta x' = -\frac{2\sqrt{\pi}\, r_e\, \Delta N_-}{S\gamma_+} \left[w(z/S) - e^{-|\zeta|^2/2} w(\eta/S)\right] \tag{15}$$

where x and y are the transverse coordinates of the positron relative to the center of the opposing electron bunch, $r_e \simeq 2.818 \times 10^{-15}$ m is the classical electron radius, γ_+ is the usual relativistic factor of the positron, and where

$$z \equiv x + iy \tag{16a}$$

$$\eta \equiv \frac{x\sigma_{y-}}{\sigma_{x-}} + i\frac{y\sigma_{x-}}{\sigma_{y-}} \tag{16b}$$

$$\zeta \equiv \frac{x}{\sigma_{x-}} + i\frac{y}{\sigma_{y-}} \tag{16c}$$

$$S \equiv \sqrt{2(\sigma_{x-}^2 - \sigma_{y-}^2)} \tag{16d}$$

In equation (15) $w(z)$ is the complex error function[1] which, for an arbitrary complex argument z, is defined by [12]

$$\int_0^z du e^{u^2} = \frac{\sqrt{\pi}}{2i}\left[e^{z^2} w(z) - 1\right] \tag{17}$$

[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.

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. For the purposes of this note, we will take four possible algorithms for the computation of $w(z)$.

2.2.1 Algorithm #1 (IMSL® function CERFE).

In this case we simply compute $w(z)$ by invoking the single-precision function CERFE(z) provided by the IMSL Math Library® v. 10.0 [13].

2.2.2 Algorithm #2 (table interpolation to 3rd order).

In this case we first generate and store a two-dimensional table of values for $w(z)$ for a uniform grid of size 0.1 in both x and y in the region $0 \le x \le 3.9$, $0 \le y \le 3.0$. This table is generated by using the IMSL® function CERFE(z), and corresponds to Table 7.9 (p. 325) of ref. [12]. In addition to the function itself, we simultaneously generate and store a table of values for each of the first four derivatives of $w(z)$ for the same grid points, obtained from the formulas [12]

$$w'(z) = -2zw(z) + \frac{2i}{\sqrt{\pi}} \tag{18a}$$

$$w^{(n+2)}(z) = -2zw^{(n+1)}(z) - 2(n+1)w^{(n)}(z), \qquad n = 0,\ 1,\ 2, \cdots \tag{18b}$$

where $w^{(0)} \equiv w(z)$ and $w^{(1)} \equiv w'(z)$.

For values of z that lie in between the points in the table, we use 3rd order interpolation: we first determine the grid point z_0 that is nearest z, and then use the truncated Taylor expansion

$$w(z) = \sum_{n=0}^{3} \frac{w^{(n)}(z_0)}{n!}\,(z - z_0)^n \tag{19}$$

If z is in the first quadrant but is outside the grid (*i.e.*, $x > 3.9$ and/or $y > 3.0$), we use the rational approximations given on p. 328 of Ref. [12].

It is not necessary to know $w(z)$ when z is not in the first quadrant because the kick $(\Delta x', \Delta y')$ (Eq. (15)) has odd-parity reflection properties. This means that $\Delta x'(-x, y) = -\Delta x'(x, y)$ and $\Delta x'(x, -y) = +\Delta x'(x, y)$, while $\Delta y'(x, y)$ has the opposite properties. These properties follow from the fact that an even-parity charge distribution (such as the bi-gaussian, Eq. (14)) gives rise to an odd-parity electric field. Thus in practice we only need to compute the kick for $|x|$ and $|y|$. The signs of $\Delta x'$ and $\Delta y'$ are always the opposite of those of x and y, respectively.

2.2.3 Algorithm #3 (table interpolation to 4th order).

This algorithm is the same as #2 with the only exception that we use 4th order interpolation (top summation limit =4 in Eq. (19)) when z is in the region covered by the grid.

2.2.4 Algorithm #4 (Padé approximant).

In this case we compute $w(z)$ by using use the Padé approximant from Ref. [14].

Table 2 displays values of $w(z)$ for the four algorithms. The three values of z we have selected belong to each of the three distinct regions used by algorithms 2 and 3. Comparisons for other values of z show equally good agreement. On the other hand, as discussed in Sec. 10, there are significant differences in the computation time used by the algorithms

Table 2: Values of $w(z)$ for three values of z.

$z =$	$1.55 + 1.55i$	$5 + 5i$	$7 + 7i$
Alg. #1	$0.1942284 + 0.1604948i$	$0.056965437 + 0.055838738i$	$0.040501639 + 0.040090580i$
Alg. #2	$0.1942282 + 0.1604948i$	$0.056965437 + 0.055838738i$	$0.040501632 + 0.040090576i$
Alg. #3	$0.1942284 + 0.1604948i$	$0.056965437 + 0.055838738i$	$0.040501632 + 0.040090576i$
Alg. #4	$0.1942285 + 0.1604948i$	$0.056965441 + 0.055838745i$	$0.040501639 + 0.040090583i$

2.3 The beam-beam kick from one slice.

In all cases described in this note we make the following assumptions: (a) The particles are relativistic so that they do not self-interact. (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.[2]

As the particles in the weak beam move through the strong beam, they encounter the N_s kicks sequentially. There are four steps that describe the algorithm for the beam-beam kick of a given particle in the weak beam with a given slice in the strong beam. These steps are the following:

Step 1: Determine the collision point. Figure 3 sketches the collision of a positron with longitudinal coordinate z_+ with a slice of the electron bunch with longitudinal coordinate z_- (both z_+ and z_- remain unchanged during the collision). 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. Therefore the azimuthal coordinates of the colliding positron (s_+) and opposing electron slice (s_-) at time t are

$$\begin{aligned} s_+ &= ct + z_+ && \text{positron} \\ s_- &= -ct - z_- && \text{electron slice} \end{aligned} \tag{20}$$

(we take the convention that $z > 0$ means the head of the bunch for either bunch; thus in Fig. 3 both z_+ and z_- are > 0 even though, when viewed from a common reference frame, the colliding particle and the slice are on opposite sides of their respective bunch centers). The collision point between the positron and the opposing slice is determined by setting $s_+ = s_- \equiv s_c$, which implies

$$s_c = \tfrac{1}{2}(z_+ - z_-) \qquad \text{collision point} \tag{21}$$

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:

$$\begin{aligned} x &\to x + s_c x' \\ y &\to y + s_c y' \end{aligned} \tag{22}$$

while the slopes x' and y' remain unchanged.

Step 3: Actual kick. In this step the slopes of the particle are changed according to Eq. (15),

$$\begin{aligned} x' &\to x' + \Delta x' \left(x, y, \sigma_{x-}(s_c), \sigma_{y-}(s_c)\right) \\ y' &\to y' + \Delta y' \left(x, y, \sigma_{x-}(s_c), \sigma_{y-}(s_c)\right) \end{aligned} \tag{23}$$

[2] The code TRS does not require assumptions (c)–(e).

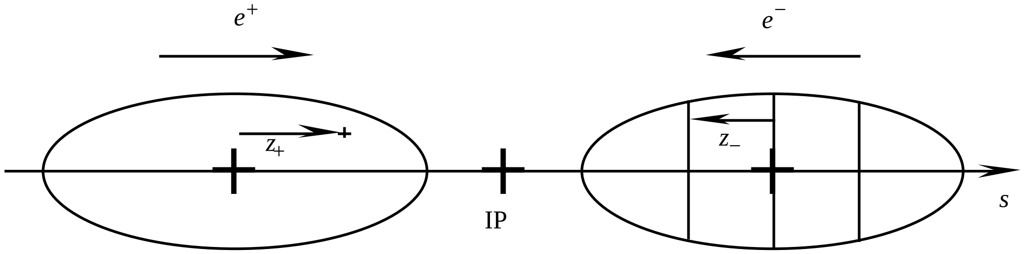

Figure 3: A positron and an electron bunch are about to collide such that their centers, with trajectories $s_\pm = \pm ct$, will come together at the IP at time $t = 0$. An individual positron with longitudinal coordinate z_+ will collide with the first electron slice, with longitudinal coordinate z_-. In the particular situation sketched, both z_+ and z_- are > 0, since they are both at the head of their respective bunches. The azimuthal coordinate s is measured from the IP and grows in the direction downstream of the positron bunch.

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

$$\begin{aligned} \sigma_{x-}(s_c) &= \sigma_{x-}(0) \times \sqrt{1 + \left(s_c/\beta^*_{x-}\right)^2} \\ \sigma_{y-}(s_c) &= \sigma_{y-}(0) \times \sqrt{1 + \left(s_c/\beta^*_{y-}\right)^2} \end{aligned} \tag{24}$$

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

$$\begin{aligned} x &\to x - s_c x' \\ y &\to y - s_c y' \end{aligned} \tag{25}$$

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.

These steps are repeated for each slice and must be carried out in the *actual sequence* of kicks encountered by the particle. It is obviously possible to save some steps in the process by concatenating Step 4 of slice k with Steps 1 and 2 of slice $k + 1$ at the expense of giving up the modularity of the algorithm. Since the CPU time used is dominated by Step 3, this concatenation is not worth the trouble.

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. It does not matter whether the loop over particles is inside or outside the loop over slices: the results are the same on account of the assumed absence of self-interaction within the dynamical bunch. However, if all the particles in the weak bunch have $z = 0$ (*e.g.*, in simulations which neglect synchrotron oscillations, damping and quantum excitation), it is computationally more efficient to loop over particles inside the loop over slices because then the actual beam sizes, Eqs. (24), need to be computed only once for each slice.

3 Transverse and longitudinal phase space maps.

In all simulations presented in this note, the machine lattice is assumed to be linear and uncoupled (radiation damping and quantum excitation effects are explained in the following section). Therefore the one-turn mapping that relates turn n to $n+1$ at a given observation point for the horizontal phase space of a given particle is given by

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{n+1} = \begin{pmatrix} C_x + \alpha_x S_x & \beta_x S_x \\ -\gamma_x S_x & C_x - \alpha_x S_x \end{pmatrix} \begin{pmatrix} x \\ x' \end{pmatrix}_n \tag{26}$$

where x and x' are the coordinate and slope of the particle, α_x, β_x and γ_x (with $\beta_x\gamma_x = 1 + \alpha_x^2$) are the lattice functions at the observation point, $C_x \equiv \cos(2\pi\nu_{0x})$, $S_x \equiv \sin(2\pi\nu_{0x})$ and ν_{0x} is the horizontal lattice tune, assumed to be an input quantity. A similar mapping applies to the vertical phase space, obtained from the above by replacing $x \to y$ everywhere. In practice, we take the observation point just before the beam-beam collision at the interaction point (IP), so that $\alpha_x = \alpha_y = 0$.

Similarly, the longitudinal phase space is subject to a simple linear transformation. Each particle is described by longitudinal dimensionless phase-space coordinates z/σ_z and $\Delta p/\sigma_p$, where z is the displacement of the particle from the center of its own bunch, Δp is its momentum deviation from the nominal momentum, and σ_p is the rms of the momentum distribution. Thus the longitudinal phase space mapping is defined by

$$\begin{pmatrix} z/\sigma_z \\ \Delta p/\sigma_p \end{pmatrix}_{n+1} = \begin{pmatrix} \cos(2\pi\nu_s) & \sin(2\pi\nu_s) \\ -\sin(2\pi\nu_s) & \cos(2\pi\nu_s) \end{pmatrix} \begin{pmatrix} z/\sigma_z \\ \Delta p/\sigma_p \end{pmatrix}_n \tag{27}$$

where ν_s is the synchrotron tune, also assumed to be an input quantity.

4 Damping and quantum excitation.

In the results presented in Secs. 7 and 8 for the one- and two-dimensional distributions (but not for the single-particle tracking results in Secs. 5 and 6), the simulations also include the effects of damping and quantum excitation. These effects are modeled by a localized stochastic map that is applied just before the beam-beam kick at the IP, where $\alpha_x = \alpha_y = 0$ [9, 15, 16, 17]. This map consists of the following transformations for every particle:

$$x \to \lambda_x x + \hat{r}_1 \sigma_x \sqrt{1-\lambda_x^2} \tag{28a}$$

$$x' \to \lambda_x x' + \hat{r}_2 \sigma'_x \sqrt{1-\lambda_x^2} \tag{28b}$$

$$y \to \lambda_y y + \hat{r}_3 \sigma_y \sqrt{1-\lambda_y^2} \tag{28c}$$

$$y' \to \lambda_y y' + \hat{r}_4 \sigma'_y \sqrt{1-\lambda_y^2} \tag{28d}$$

$$z \to \lambda_z z + \hat{r}_5 \sigma_z \sqrt{1-\lambda_z^2} \tag{28e}$$

$$\Delta p \to \lambda_z \Delta p + \hat{r}_6 \sigma_p \sqrt{1-\lambda_z^2} \tag{28f}$$

where the σ's are taken as given input quantities (the transverse σ's are evaluated at the observation point).

The first term in each case represents the radiation damping, and the second term the quantum excitation. The λ's are given in terms of the damping times τ (measured in units of turns) by

$$\lambda_i \equiv e^{-1/\tau_i}, \qquad i = x, y, z \tag{29}$$

The τ's are taken as given input parameters in our calculations, and are listed in Table 10 below. As is well known, however, these are not independent: they are related to the relative synchrotron radiation energy loss per turn $\delta \equiv \Delta E/E$ and the damping partition numbers J_i by $\tau_i^{-1} = \delta J_i/2$ with $J_x + J_y + J_z = 4$ [18]. For an isomagnetic lattice, the J_i's are further constrained by $J_x = 1 - \mathcal{D}$, $J_y = 1$ and $J_z = 2 + \mathcal{D}$ so that $\tau_x^{-1} + \tau_z^{-1} = 3\tau_y^{-1}$.

The σ's in Eq. (28) are the nominal beam sizes of the dynamical beam, also assumed to be input parameters in the calculations. The $\hat{r}_i$'s are independent random numbers with the properties

$$\langle \hat{r}_i \rangle = 0 \quad \text{and} \quad \langle \hat{r}_i \hat{r}_j \rangle = \delta_{ij} \tag{30}$$

Obviously, there is an infinite number of distributions for the $\hat{r}$'s that have these properties. In the codes discussed in this note, we use two possibilities:

- **Algorithm #1**: $\hat{r}$ is derived from a normal distribution with zero mean and unit standard deviation.
- **Algorithm #2**: $\hat{r}$ is derived from a uniform distribution in the interval $[-\sqrt{3},\, +\sqrt{3}]$

It can be easily shown that, in the absence of the beam-beam interaction, the combined maps for the linear transport (Eq. (26)) plus synchrotron rotation (Eq. (27)) plus damping and quantum excitation (Eq. (28)) yield a particle distribution whose core is an upright gaussian matched to the lattice, with the specified rms sizes σ_x, σ_y and σ_z. The equilibrium state is reached as the number of turns $\to \infty$ although, in practice, it is sensibly reached in a few damping times. The equilibrium distribution is independent of the initial conditions and, as a consequence of the central limit theorem, its *core* is independent of the algorithm used for the $\hat{r}$'s provided it satisfies the conditions (30). On the other hand, the *tail* of the equilibrium distribution (normalized amplitudes larger than $\sim \delta^{-1}$) is sensitive [19] to the algorithm used to generate the $\hat{r}$'s. In this note, however, we will not be concerned with such large amplitudes.

In the calculations presented in Secs. 7 and 8 (which do take into account radiation damping and quantum excitation), the codes LIFETRAC and TAIL use all six Eqs. (28) and algorithm #1 for the $\hat{r}$'s. The corresponding calculations with TRS represent radiation damping and quantum excitation by Eqs. (28) except that (28e) is replaced by the identity map, $z \to z$, and the $\hat{r}$'s are generated according to algorithm #2. In the absence of the beam-beam kick, these maps converge to the same steady-state distribution (strictly speaking, in the map used in TRS the convergence requires $\nu_s \neq$ integer).

5 Short-term single-particle tracking results.

In this first set of exercises 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.

As mentioned earlier, we carry out a weak-strong calculation where the dynamical (or weak) beam is 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 data is obtained as follows: we first assign input values to all six normalized coordinates. The particle then undergoes a linear lattice transport (Eq. (26)), then a synchrotron rotation (Eq. (27)), 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 in the first line of a disk file as "turn 1." This process is then repeated for N turns, with one line of output per turn.

The tracking data is processed with the code PORTRAIT [20], from which we obtain all three normalized phase spaces and their corresponding Fourier spectra. Each spectrum is produced as follows: we first compute

$$\tilde{X}_k = \sum_{n=1}^{N} X_n \, e^{2\pi i (n-1)k/N}, \qquad k = 0,\, 1,\, \cdots,\, N-1 \tag{31}$$

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

Table 3: Parameters used in the simulations.

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

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

Table 4: Initial conditions for tracking.

x_0	x_0'	y_0	y_0'	z_0	Δp_0
$3\sigma_x$	0	$1.5\sigma_y$	0	$3\sigma_z$	0

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[3] 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 similar to those considered for the PEP-II B factory [1]; 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}. In all cases in this section we track for $N = 512$ turns and we use $N_s = 5$ kicks. The initial values of the coordinates are always the same and are listed in Table 4.

The results shown in Figs. 4 through 16 are a small subset of cases we have studied. The agreement in the comparisons shown in these figures is typical of the larger set.

5.1 Comparison of the five slicing algorithms.

We first carry out a comparison for the five slicing algorithms described in Sec. 2.1. Results are obtained by tracking a positron for $N = 512$ turns with the code BBTRACK3D [5]. Table 5 lists the simulation conditions and the figure numbers that correspond to each set of results. One can see from the figures that

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

the differences are perceptible to the eye only in the lower end of the vertical spectrum.

Table 5: Comparisons for the five slicing algorithms.

Fig. #	code	N_s	slicing alg. #	$w(z)$ alg. #
4	BBTRACK3D	5	1	1
5	BBTRACK3D	5	2	1
6	BBTRACK3D	5	3	1
7	BBTRACK3D	5	4	1
8	BBTRACK3D	5	5	1

5.2 Comparison of the four algorithms for $w(z)$.

Next we carry out a comparison of the four algorithms for $w(z)$ described in Sec. 2.2. We track with the code BBTRACK3D for $N = 512$ turns, use $N_s = 5$ kicks, and keep the slicing algorithm fixed (#2, equal weights). Table 6 lists the figures that correspond to each case.

Table 6: Comparisons for the four algorithms for $w(z)$.

Fig. #	code	N_s	slicing alg. #	$w(z)$ alg. #
9	BBTRACK3D	5	2	1
10	BBTRACK3D	5	2	2
11	BBTRACK3D	5	2	3
12	BBTRACK3D	5	2	4

5.3 Comparison of the four codes.

Finally, we compare the results of all 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 $w(z)$ with the IMSL® library (algorithm #1) except for the code TAIL, in which we use the Padé approximant (algorithm #4). Table 7 lists the information.

Table 7: Comparisons for the four codes.

Fig. #	code	N_s	slicing alg. #	$w(z)$ alg. #
13	BBTRACK3D	5	2	1
14	TRS	5	2	1
15	LIFETRAC	5	2	1
16	TAIL	5	2	4

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).

5.4 Effects of deliberate errors.

In order to get an idea of the effects of nontrivial algorithmic errors, we show in Figs. 17 and 18 the results arising from incorrect coding. Figure 17 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. (23)) was coded as

$$\left.\begin{aligned} x' &\to x' + \left(\beta_{x-}(s_c)/\beta^*_{x-}\right)\Delta x' \\ y' &\to y' + \left(\beta_{y-}(s_c)/\beta^*_{y-}\right)\Delta y' \end{aligned}\right\} \quad \text{incorrect!} \tag{32}$$

Other than this difference, all tracking conditions used in producing the results shown in Fig. 17 were identical to those used in Fig. 14. By comparing Fig. 17 with the correct results, shown again in Fig. 19, one can see clear differences: the vertical amplitudes reach out to ∼ 3 in Fig. 17 rather than to ∼ 1.5 in Fig. 14, and the vertical spectra are substantially different.

As discussed in Sec. 2, Step 4 ("disruption") of the algorithm for the thick lens beam-beam interaction makes the sequence of kicks experienced by the tracked particle noncommutative. Figure 18 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 Fig. 19. The phase spaces are not substantially different, but the vertical spectra are clearly different at the low end.

It turns out that errors like these lead to only minor effects in multiparticle simulations such as those carried out for PEP-II [1]. However, they might be more important in other situations. It is difficult to uncover these kinds errors, particularly in those codes that embody more complicated physical models. One of our main motivations in carrying out the detailed single-particle comparisons was to look for these. It is gratifying that the spectrum of the motion provides such a useful "magnifying glass" through which to look at the beam-beam interaction.

6 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.

6.1 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. [21], except for one detail which we will clarify below. The first step is to recall that the electromagnetic kick from one slice, Eq. (15), can be written as a two-dimensional vector as

$$(\Delta x', \Delta y') = -\frac{r_e\, N_-}{2\gamma_+}\left(\mathbf{E} + \mathbf{v}_+ \times \mathbf{B}\right) = -\frac{r_e\, N_-}{\gamma_+}\,\mathbf{E} \tag{33}$$

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 the equation

$$\nabla \cdot \mathbf{E} = 4\pi\rho_t \tag{34}$$

where $\rho_t(x, y, \sigma_{x-}, \sigma_{y-})$ is the transverse gaussian density, Eq. (14) (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). Now we can define the potential V as

$$\mathbf{E} = -\nabla V \tag{35}$$

and it is straightforward to show that V is given[4] by [22, 23]

$$V(x,y) = \int_0^\infty dt \, \frac{\exp\left[-\frac{1}{2}\left(\frac{x^2}{t+\sigma_{x-}^2} + \frac{y^2}{t+\sigma_{y-}^2}\right)\right]}{\sqrt{(t+\sigma_{x-}^2)(t+\sigma_{y-}^2)}} \tag{36}$$

The "phase averaged beam-beam parameter" of the kicked positron is given by [21]

$$\xi_i = -\frac{r_e\, N_-}{2\pi\gamma_+}\frac{\partial T_{00}}{\partial I_i}, \qquad i = x, y \tag{37}$$

where T_{00} is defined by

$$T_{00} = \int_0^{2\pi} \frac{d\theta_x}{2\pi} \int_0^{2\pi} \frac{d\theta_y}{2\pi}\, V\Big(\sqrt{2\beta_{x+}^* I_x}\cos\theta_x, \sqrt{2\beta_{y+}^* I_y}\cos\theta_y\Big) \tag{38a}$$

$$\equiv \Big\langle V\left(\sqrt{2\beta_{x+}^* I_x}\cos\theta_x, \sqrt{2\beta_{y+}^* I_y}\cos\theta_y\right)\Big\rangle \tag{38b}$$

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

$$\cos\left(2\pi(\nu_{0i} + \Delta\nu_i)\right) = \cos(2\pi\nu_{0i}) - 2\pi\xi_i \sin(2\pi\nu_{0i}), \qquad i = x, y \tag{39}$$

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

In Ref. [21] the ξ's are assumed to be small enough that the approximation $\Delta\nu_i = \xi_i$ (which follows from Eq. (39) for small enough ξ) is valid. Hence in this approximation $\Delta\nu_i$ is given directly by Eq. (37). Our approach, which involves the extra step (39), 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.*[5]

It should be noted that $\sqrt{2\beta_{x+}^* I_x}$ and $\sqrt{2\beta_{y+}^* I_y}$ in Eq. (38) 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

$$\frac{\partial}{\partial I_y} = \frac{\beta_{y+}^*}{y_0}\frac{\partial}{\partial y_0} \tag{40}$$

we obtain

$$\xi_y = -\frac{r_e\, N_- \beta_{y+}^*}{2\pi\gamma_+\, y_0}\frac{\partial}{\partial y_0}\left\langle V\left(x_0\cos\theta_x, y_0\cos\theta_y\right)\right\rangle \tag{41a}$$

$$= +\frac{r_e\, N_- \beta_{y+}^*}{2\pi\gamma_+\, y_0}\left\langle \cos\theta_y\, E_y\left(x_0\cos\theta_x, y_0\cos\theta_y\right)\right\rangle \tag{41b}$$

$$= -\frac{\beta_{y+}^*}{2\pi y_0}\left\langle \cos\theta_y \Delta y'\right\rangle \tag{41c}$$

$$= \left\langle 2\cos^2\theta_y \xi_{y\ell}\left(x_0\cos\theta_x, y_0\cos\theta_y\right)\right\rangle \tag{41d}$$

[4] The well-known logarithmic ("infrared") divergence of the integral at $t \sim \infty$ has no physical consequence.

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

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

$$\Delta y'(x,y) \equiv -4\pi\xi_{y\ell}(x,y)\frac{y}{\beta^*_{y+}} \tag{42}$$

The phase integrals above can be carried out [24] and the result expressed in terms of the modified Bessel function I_0 by using the formula [12]

$$\int_0^{2\pi} \frac{d\theta}{2\pi} e^{-2z\cos^2\theta} = e^{-z} I_0(z) \tag{43}$$

valid for arbitrary z (real or complex). However, there is an integral over t left over from Eq. (36) that, in most cases, can only be done numerically. In these cases it is far more convenient to use Eqs. (41b) or (41c) than (41a).

As a simple check, the well-known expression for the low-amplitude beam-beam parameter can be recovered from Eq.(41) by using the expression for the electric field seen by a positron near the axis,

$$E_y = \frac{2y}{\sigma_{y-}(\sigma_{x-}+\sigma_{y-})} \qquad (x, y \text{ small}) \tag{44}$$

so that Eq.(41) yields

$$\xi_y = \frac{r_e\, N_- \beta^*_{y+}}{2\pi\gamma_+\, y_0} \frac{\langle 2y_0 \cos^2\theta_y\rangle}{\sigma_{y-}(\sigma_{x-}+\sigma_{y-})} \tag{45a}$$

$$= \frac{r_e\, N_- \beta^*_{y+}}{2\pi\gamma_+\, \sigma_{y-}(\sigma_{x-}+\sigma_{y-})} \tag{45b}$$

6.1.1 Round beam case.

An important special case is that of round beams, in which case the integrals in Eq. (41) can be done in closed form [25]. In this case $\sigma_{x-} = \sigma_{y-} \equiv \sigma_-$, and the electric field is given by

$$\mathbf{E} = \frac{2\mathbf{x}}{\mathbf{x}^2}\left(1 - e^{-\mathbf{x}^2/2\sigma_-^2}\right) \tag{46a}$$

$$= \frac{\mathbf{x}}{\sigma_-^2}\int_0^1 dt\, e^{-t\mathbf{x}^2/2\sigma_-^2} \tag{46b}$$

Let us take the special case in which we choose $x_0 = 0$ and focus our attention on the vertical tune shift; since the strong beam is round, the phase-averaged beam-beam parameter of the positron must depend only on the radial distance from the beam center. By inserting the above expression into Eq. (41b) we obtain

$$\xi_y = \frac{r_e\, N_- \beta^*_{y+}}{2\pi\gamma_+\, \sigma_-^2}\int_0^1 dt \int_0^{2\pi} \frac{d\theta_y}{2\pi}\cos^2\theta_y\, e^{-2vt\cos^2\theta_y} \tag{47a}$$

$$= \frac{r_e\, N_- \beta^*_{y+}}{2\pi\gamma_+\, \sigma_-^2}\left(\frac{-1}{2v}\right)\int_0^1 dt \frac{\partial}{\partial t}\left(e^{-vt} I_0(vt)\right) \tag{47b}$$

$$= \frac{\xi_{0y}}{v}\left[1 - e^{-v} I_0(v)\right] \tag{47c}$$

where we have defined $v \equiv (y_0/2\sigma_-)^2$ and where ξ_{0y} is the nominal (zero amplitude) beam-beam parameter of the positron in collision with a round electron beam,

$$\xi_{0y} = \frac{r_e\, N_- \beta^*_{y+}}{4\pi\gamma_+\, \sigma_-^2} \tag{48}$$

Actually, expression (47c) can be obtained in a simpler way directly from Eq. (41a) by noting that, for a round beam and for $x_0 = 0$, the phased-averaged potential is

$$\langle V(0, y_0 \cos\theta_y)\rangle = \int\limits_0^1 \frac{du}{u} e^{-vu} I_0(vu) \tag{49}$$

which is obtained from Eq. (36) by the change of variables $u = \sigma_-^2/(t+\sigma_-^2)$ (the logarithmic divergence at $t \sim \infty$ reappears here at $u \sim 0$). Upon taking $\partial/\partial y_0$ and using standard properties of the modified Bessel function, one obtains the result (47c).

We have carried out single-particle tracking for a round beam model with the code TRS and computed the tune with PORTRAIT. The lattice parameters are listed in Table 8). We use one kick ($N_s = 1$), we track for $N = 512$ turns, and we use the IMSL® library to compute $w(z)$. Radiation damping and quantum excitation are wholly neglected. The tracked particle is injected with $x_0 = z_0 = 0$, and the vertical amplitude is varied in the range $0 \le y_0 \le 10\sigma$.

The analytic calculation is done by inserting the formula for the beam-beam parameter, Eq. (47c), into Eq. (39) and solving for $\Delta\nu_y$. The tracking results (crosses) are shown in Fig. 20 along with a plot of the analytic calculation (solid). The agreement is almost perfect.

Table 8: Parameters for round-beam tracking.

	e^+	e^-
E [GeV]	5.3	5.3
β_x^* [m]	0.05	0.05
β_y^* [m]	0.05	0.05
σ_x [μm]	100	100
σ_y [μm]	100	100
ν_{0x}	0.64	0.64
ν_{0y}	0.64	0.64
N	2.78×10^{11}	2.78×10^{11}
ξ_{0x}	0.03	0.03
ξ_{0y}	0.03	0.03

6.1.2 Flat beam case.

In this case 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 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. (41c). Results are shown in Fig. 21. Again, the agreement is almost perfect.

6.2 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. (21)) 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

$$\Delta y'(x, y, s_c) = -\frac{r_e N_-}{\gamma_+} E_y(x, y, \sigma_{x-}(s_c), \sigma_{y-}(s_c)) \tag{50}$$

and similarly for $\Delta x'$. Here E_y is given by Eq. (15) with the only proviso that $\sigma_{x-}(s_c)$ and $\sigma_{y-}(s_c)$ are the actual beam sizes of the electron bunch at the collision point, given by Eq. (24). Therefore it seems clear that the generalization to the present case of the phased-averaged vertical beam-beam parameter is

$$\xi_y = \left\langle 2\cos^2\theta_y\, \xi_{y\ell}\,(x_0\cos\theta_x, y_0\cos\theta_y, (z_0/2)\cos\theta_z)\right\rangle \tag{51}$$

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

$$\Delta y'(x, y, s_c) \equiv -4\pi\xi_{y\ell}(x, y, s_c)\frac{y}{\beta_{y+}(s_c)} \tag{52}$$

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

$$\langle\cdots\rangle = \int_0^{2\pi}\frac{d\theta_x}{2\pi}\int_0^{2\pi}\frac{d\theta_y}{2\pi}\int_0^{2\pi}\frac{d\theta_z}{2\pi}(\cdots) \tag{53}$$

and that the beta function in Eq. (52) has the appropriate s_c-dependence.

6.2.1 Case when the positron is close to the axis.

Fig. 22 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 $w(z)$. 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. 22 correspond to different ratios of the beta functions of the two beams (in all 3 cases, however, the beams are flat). 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 9 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 [26]. 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 that $\xi_{y+}(z_0)$ scales like [26]

$$\xi_{y+}(z_0) \sim \frac{\beta_{y+}(z_0/2)}{\sqrt{\beta_{y-}(z_0/2)}} \tag{54}$$

Table 9: 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

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 9, the tune shift is larger the larger is β_{y-}^*. Actually, if this scaling formula (properly normalized) is inserted into Eq. (39), the resultant vertical tune is in good qualitative agreement with the more accurate calculations shown in Fig. 22.

For the high asymmetry case, Fig. 22 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.

6.2.2 Case when the positron is away from the axis.

Finally, Fig. 23 shows the vertical tune of a positron oscillating vertically and longitudinally at the same time. The calculation was again done with TRS for 512 turns, IMSL® method for the calculation of $w(z)$, and $N_s = 1$. Damping and quantum excitation were ignored. The tracking conditions are those corresponding to the "nominal" case (Table 9, middle column). The positron was launched with $x_0 = x_0' = y_0' = \Delta p_0 = 0$ in all cases, and the vertical tune is plotted against the normalized vertical launching amplitude. The six sets of data correspond to six different values for the longitudinal amplitude z_0.

The lowest curve, corresponding to $z_0 = 0$, is exactly the same as that shown in Fig. 21. In this case, tracking and analytic results agree excellently. When $z_0 > 0$, however, the tracking data agrees with the analytic curves very well only for small values of y_0. In fact, the tracking data do not follow a smooth curve. An analysis with PORTRAIT shows that the synchrotron sidebands $\nu_y \pm \nu_s$ are prominent in this region of the amplitude space. This is clearly seen in Fig. 24, which shows the phase space and spectrum of the particle launched with $x_0 = 0,\ y_0/\sigma_y = 2,\ z_0/\sigma_z = 3$. It is virtually certain that these sidebands are responsible for the lack of agreement between the data and the analytic calculation (the analytic calculation is insensitive to these kinds of resonances, hence the smooth curves).

Table 10: Parameters used in the simulations for the density.

	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
τ_x [turns]	5400	5014
τ_y [turns]	5400	5014
τ_z [turns]	2700	2507

7 Medium-term tracking: 1-D amplitude distributions.

In this section we present the results in the form of particle density plotted as a function of normalized amplitude squared. The one-dimensional horizontal density is defined by

$$\frac{1}{N_0}\frac{dN}{d\hat{A}_x^2}, \quad \text{with} \quad \hat{A}_x^2 \equiv \frac{x^2 + (\beta_x x' + \alpha_x x)^2}{\sigma_x^2} \tag{55}$$

with corresponding expressions for the vertical counterparts. Here N represents the number of particle-turns at 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. Note that, by definition, the densities are normalized to unity, namely

$$\int_0^\infty d\hat{A}^2 \, \frac{1}{N_0}\frac{dN}{d\hat{A}^2} = 1 \tag{56}$$

for both x and y.

In this kind of simulation, done with the code TRS [2], we do need to include damping and quantum excitation. The algorithms for these are given by Eqs. (28) except that Eq. (28e) is ignored altogether (*i.e.*, it is replaced by the identity, $z \to z$). The random numbers $\hat{r}$ are generated according to algorithm #2. The beam parameters we use are those from Table 3 augmented by the damping times; the complete set is displayed in Table 10.

We obtain the densities as follows: we start 1024 representative particles of the weak beam with an initial nominal gaussian distribution in 6-D phase space. We track them for 5000 turns (*i.e.*, roughly one damping time), and then record their horizontal and vertical normalized amplitudes at every turn for the succeeding 5000 turns; thus we accumulate a total of $N_0 = 5.12 \times 10^6$ particle-turns. We then bin the $\hat{A}^2$'s into 500 equal-size bins in the range $0 \le \hat{A}^2 \le 25$ for either x and y, and plot the resultant histograms.

For reference we note that, in the absence of the beam-beam interaction, the expected result for the density is the "nominal gaussian" distribution, given by

$$\frac{1}{N_0}\frac{dN}{d\hat{A}^2}=\frac{1}{2}e^{-\hat{A}^2/2} \qquad \text{(nominal gaussian)} \tag{57}$$

for either x or y. If the beam-beam parameter is small (but nonzero), the expected result for the phase space distribution is not the nominal gaussian but rather a gaussian matched to the "dynamical beta-function" [27]. Because of this dynamical effect, the particle density, which is the projection of the phase space distribution onto the *nominal* (rather than dynamical) amplitude axis, deviates from a gaussian. For our parameter values, however, this deviation is not significant [27], and therefore we will use for reference the nominal gaussian density, Eq. (57).

7.1 Comparisons with different slicing algorithms.

Figure 25 shows the horizontal and vertical particle distributions obtained with all five slicing algorithms. Five kicks are used in all three cases. One can see that the algorithms are in good agreement with each other, although algorithm #1 yields slightly larger values for the horizontal density at large amplitudes. One also sees that, as is typical in beam-beam simulations, the horizontal distribution remains closer to the gaussian shape until larger amplitudes than the vertical distribution. This qualitative feature is generally valid in most (though not all) of the tune plane. Each run (for each algorithm) takes $\sim$ 23 CPU minutes on the Cray-2S/8128 at NERSC.

7.2 Comparisons with different number of slices.

Figure 26 show the comparisons of similar runs for different number of slices for slicing algorithm #2. In the vertical distribution one sees that there is significant change when going from 1 to 3 kicks but not when one increases the number of kicks beyond 3. The horizontal distribution shows greater sensitivity to the number of kicks at large amplitude, but since the statistics in this region are poor, one cannot extract definite conclusions.

8 Long-term tracking: 2-D amplitude distributions.

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

$$\frac{1}{N_0}\frac{dN}{d\hat{A}_x d\hat{A}_y} \tag{58}$$

which is also normalized to unity,

$$\int\limits_0^\infty d\hat{A}_x\int\limits_0^\infty d\hat{A}_y\;\frac{1}{N_0}\frac{dN}{d\hat{A}_x d\hat{A}_y}=1 \tag{59}$$

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 example, the nominal gaussian distribution for the 2-D case is

$$\frac{1}{N_0}\frac{dN}{d\hat{A}_x d\hat{A}_y}=\hat{A}_x\hat{A}_y\,e^{-(\hat{A}_x^2+\hat{A}_y^2)/2} \qquad \text{(nominal gaussian)} \tag{60}$$

whose contour plot has a characteristic scallop shape and is shown in Fig. 27. The maximum density is $1/e=0.3679$ at $\hat{A}_x=\hat{A}_y=1$. We have chosen the first contour level at a height $1/\sqrt{e}$ below the peak; successive contour levels are at a constant ratio e below each other.

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 10. The full list of parameters is displayed in Table 11.

Table 11: Parameters used in the simulations for the 2-D distributions.

	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	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 the conditions are not identical for the three codes: (1) TAIL uses the Padé approximant method to calculate the complex error function rather than the IMSL® library; (2) TRS excludes step (28e) from the algorithm for the radiation damping and quantum excitation map; (3) TRS uses algorithm #2 for the generation of the random numbers $\hat{r}$ in (28); and (4) LIFETRAC uses slicing algorithm #5 rather than #2. The full list of simulation conditions are shown in Table 12 ("RQE" stands for radiation and quantum excitation).

Table 12: Simulation conditions for 2-D distributions.

code	N_s	slicing alg. #	$w(z)$ alg. #	RQE alg.	$\hat{r}$ alg. #
TRS	5	2	1	(28) excl. (28e)	2
LIFETRAC	5	5	1	(28a)–(28f)	1
TAIL	5	2	4	(28a)–(28f)	1

Figure 28 shows the contour plots for the resultant simulation from all three codes. We use the same contour levels as in Fig. 27, namely the first contour 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.

As mentioned in the introduction, 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, the codes LIFETRAC [3] and TAIL [4] use similar (but not identical) acceleration algorithms that enhance the speed of the development of the tails of the distribution [6]. For the simulation shown in Fig. 28, 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.

9 A systematic study of the five slicing algorithms.

9.1 Convergence rate as $N_s \to \infty$.

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 depends on many variables of the problem at hand, such as the working point, beam aspect ratios, etc. A related problem, which is more tractable, is to determine how many kicks within a given algorithm is sufficient to yield a solution with acceptable accuracy for a given problem.

In this Section we try to solve a more modest problem: we study the convergence rate of the five slicing algorithms presented in Sec. 2.1 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 below an answer to the important practical question of how many kicks is enough for a given case.

We proceed as follows: we first generate a weak bunch of 1000 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 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 (as explained in Sec. 2.3, the transverse coordinates are referred to the bunch center). 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 parameters; 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,

$$Q = \sum_{n=1}^{4} \sqrt{\langle (X_n - X_{n,\infty})^2 \rangle} \tag{61}$$

where $\langle \cdots \rangle$ is the average over the 1,000 particles. Obviously, the smaller Q the better the slicing algorithm. We consider three cases, corresponding to different IP and beam parameters: round beam, flat beam (PEP-II) and flat beam (VEPP-4). The relevant parameters are listed in Table 13.

Figures 29 and 30 show the results of plotting Q *vs.*the number of kicks for all five algorithms in question. It is apparent that algorithm #4 has systematically the fastest convergence rate of the five. For large number of kicks (∼ 100), Q reaches roughly the same range of values for all five algorithms. For small number of kicks, there are more significant differences. For the round beam case, the value of Q is relatively high probably because the bunch length is relatively high as is the beam-beam parameter (see Table 13). It is curious that algorithm #1 does not converge uniformly, although it becomes competitive with #4 for $\gtrsim 50$ or more kicks. It is also interesting that algorithm #5 crosses over with #4 for the round-beam case for ∼ 20 kicks.

Table 13: Parameters used in the tests of the convergence rate of the slicing algorithms.

	Round beam (Fig. 29)	PEP-II (Fig. 30)	VEPP-4 (Fig. 30)
β^*_{x+}/σ_{z+}, β^*_{y+}/σ_{z+}	1, 1	37.5, 1.5	10, 1
$\beta^*_{x-}/\beta^*_{x+}$, $\beta^*_{y-}/\beta^*_{y+}$	1, 1	1.333, 1.333	1, 1
σ_{x-}/σ_{x+}, σ_{y-}/σ_{y+}, σ_{z-}/σ_{z+}	1, 1, 1	1, 1, 1	1, 1, 1
σ_{x-}/σ_{y-}	1	25	80
ξ_{x+}, ξ_{y+}	0.1, 0.1	0.03, 0.03	0.015, 0.06

9.2 A criterion for the adequate number of kicks.

How many kicks is enough for a given problem? This is an important practical question because, all other things being equal, the CPU time 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 establishing 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

$$\frac{\delta\sigma}{\sigma} \simeq \frac{1}{\sqrt{\tau}} \tag{62}$$

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

$$Q \simeq \frac{4}{\sqrt{\tau}} \tag{63}$$

(the factor 4 accounts for the four terms in Eq. (61)). 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. (63).

For the case of PEP-II, with $\tau = 5400$, this yields $Q \simeq 0.05$. Therefore, as one can see from Fig. 30, any of the five algorithms gives the correct answer for $N_s = 3$. This estimate is consistent with the results for the 1-D distributions shown in Fig. 26. On the other hand, for the round-beam case shown in Fig. 29, this criterion would imply (assuming, for the sake of this example, that τ is also 5400) that algorithms #3 and #4 (especially #4) would yield the correct answer with $N_s = 3$, while algorithms #2 and #5 would require $N_s = 5$, and algorithm #1 would require $N_s \gtrsim 15$, for a correct answer.

10 Conclusions.

The single-particle tracking results, for thin or thick beam-beam lens, show that the four codes under comparison are in agreement within computer accuracy when restricted to use the same algorithms. Actually the code TAIL uses the Padé approximant method for the computation of $w(z)$, which is different from the other three codes; however, since the runs with BBTRACK3D show that there is essentially no difference in the results when using the Pade approximant algorithm or any of the other 3 algorithms, the comparison among the four codes is meaningful, and the almost perfect agreement is heartening.

There is a significant difference in computational speed in the algorithms for the complex error function: a comparison carried out for the 1-D particle distributions with TRS (not shown in this note) shows that the CPU times used by algorithms #1, #2, #3 and #4 for $w(z)$ are in the ratio 14 : 1 : 1.01 : 1.3 respectively. Thus the table interpolation algorithm is clearly advantageous from the point of view of speed, although, of course, it uses more memory.

The five slicing algorithms yield almost identical results for single-particle tracking near the core of the beam. However, the 1-D distributions at large amplitude exhibit clear differences between slicing algorithms #1 (equal-spacing) and #2 (equal-weight), and small differences between #2 and #3. It is likely that these differences would be enhanced at larger values of the beam-beam parameter.

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, we clearly identified synchrotron sidebands of the vertical tune. Since the analytic calculation does not take these sidebands into account, the discrepancy is not meaningful.

For slicing algorithm #2, the one-dimensional particle distributions in amplitude show clear differences when one increases the number of kicks from 1 to 3, but not beyond 3, at least not for $\xi_0 = 0.03$. Higher values of ξ_0 and/or different slicing algorithms require more kicks for a stable answer.

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 exact because the codes use different algorithms for slicing, radiation damping and quantum excitation.

A study of the convergence rate as $N_s \to \infty$ 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 adequate 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.

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.

11 Acknowledgments.

We are grateful to NERSC for supercomputer support.

References

[1] "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.

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

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

[4] 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.

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

[6] 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.

[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] *Handbook of Mathematical Functions*, M. Abramowitz and I. A. Stegun, Eds., Dover Publications Inc., 9th printing, 1970.

[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] K. Hirata and F. Ruggiero, “Treatment of Radiation in Electron Storage Rings,” LEP Note 611, August 8, 1988.

[16] R. Siemann and S. Krishnagopal, “A Simulation Study of Radiation Treatment with a Quadrupolar Beam-Beam Kick,” CBN 88–1.

[17] S. Krishnagopal and R. Siemann, “Some Aspects of the Two-Beam Performance of DCI,” NIMPR **A313** (1992) 328-336.

[18] M. Sands, “The Physics of Electron Storage Rings: An Introduction,” SLAC-121/UC-28 (ACC), November 1970; addendum, May 1979.

[19] M. A. Furman, “On the Fixed Point of Lumped Linear Stochastic Maps” (to be documented).

[20] M. A. Furman, “The Tracking Analysis Program PORTRAIT” (to be documented).

[21] 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.)

[22] O. D. Kellogg, *Foundations of Potential Theory*, J. Springer, 1929 (reprinted by Dover Publications Inc., 1953).

[23] M. A. Furman, “Practical Approximations for the Electric Field of Elliptical Beams,” SSC-N-312, April, 1987.

[24] See, for example, A. Chao, “Beam-Beam Instability,” 1983 Summer School on High Energy Particle Accelerators, BNL/SUNY, July 6–16, 1983, AIP Conf. Proc. **127** p. 201 (M. Month, P. F. Dahl and M. Dienes, eds.).

[25] See, for example, F. M. Izraelev, “Nearly Linear Mappings and their Applications,” Physica **1D**, 243–266 (1980).

[26] 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.

[27] M. A. Furman, “The Particle Density of a Gaussian Beam Matched to the Dynamical Beta Function” CBP Tech Note 61/PEP-II/AP Note 16-94, September 16, 1994.

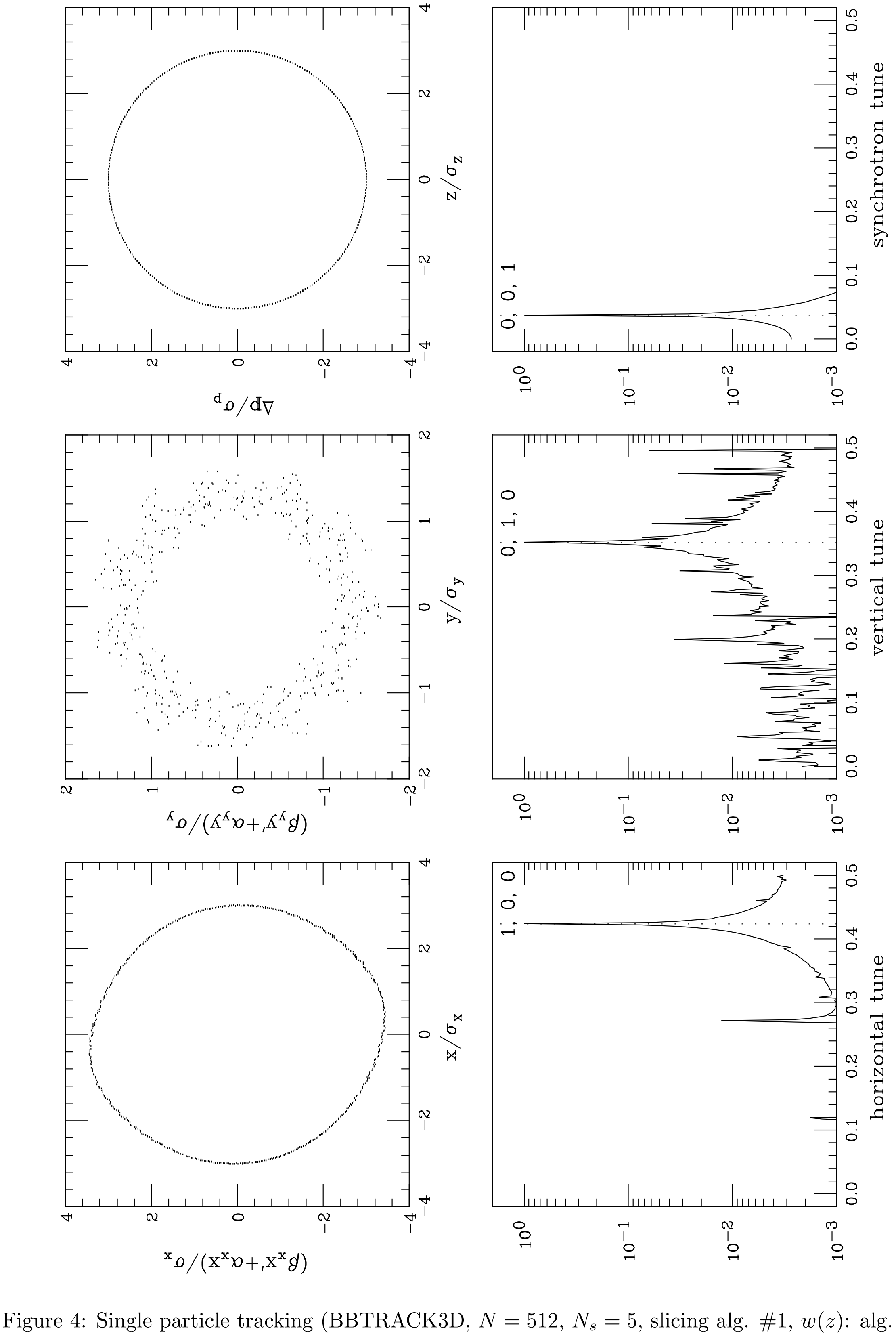

Figure 4: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #1, $w(z)$: alg. #1).

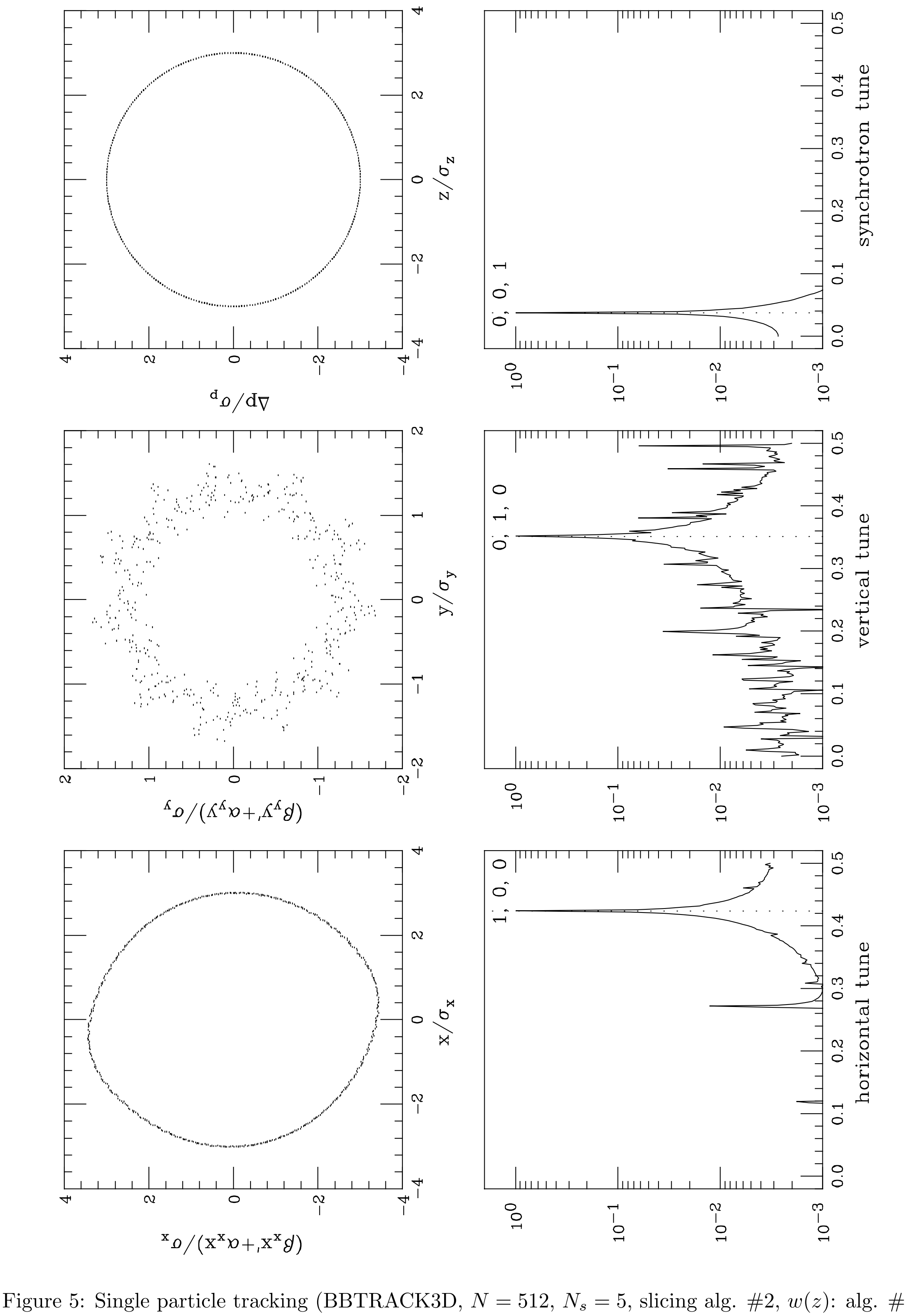

Figure 5: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1).

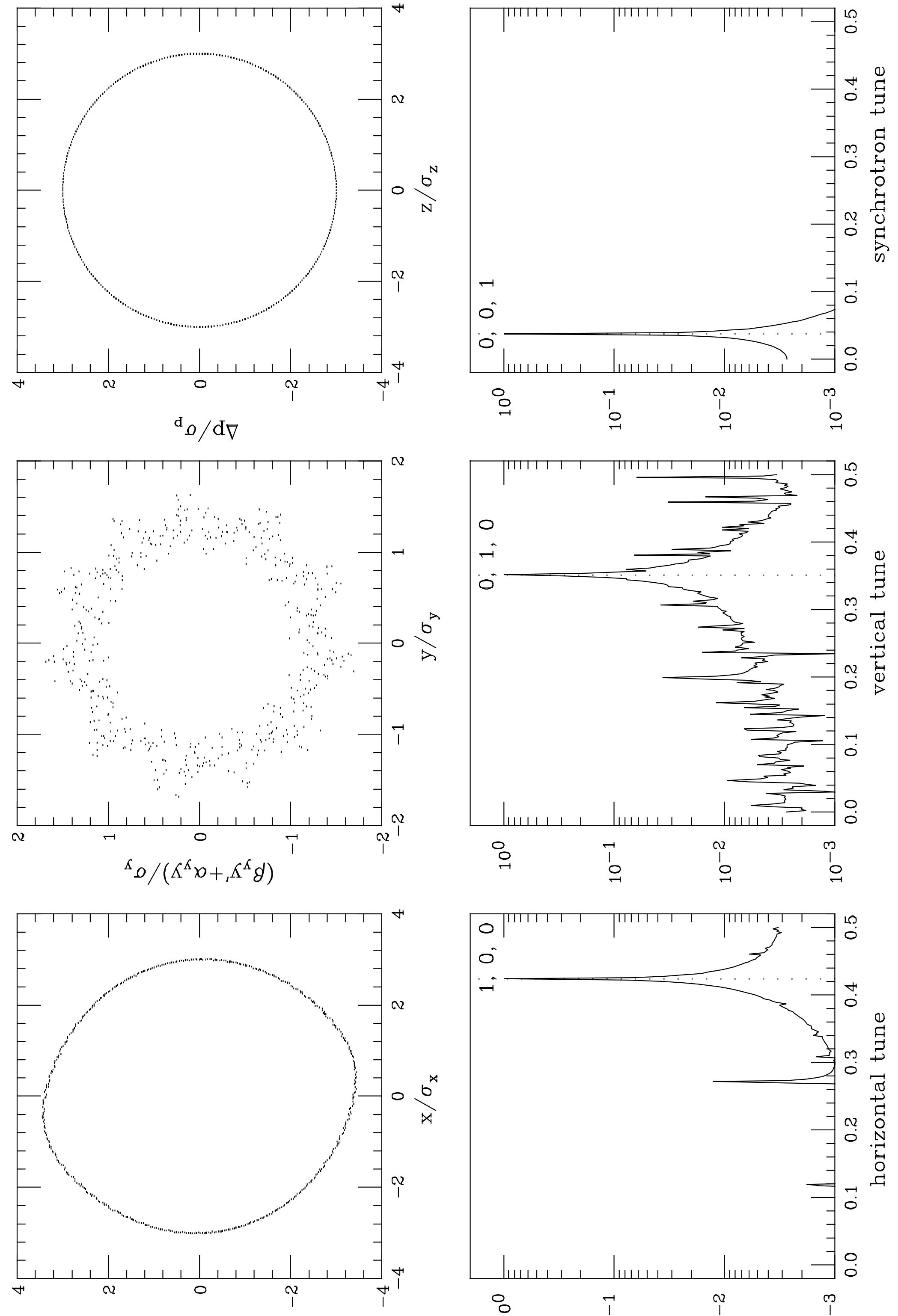

Figure 6: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #3, $w(z)$: alg. #1).

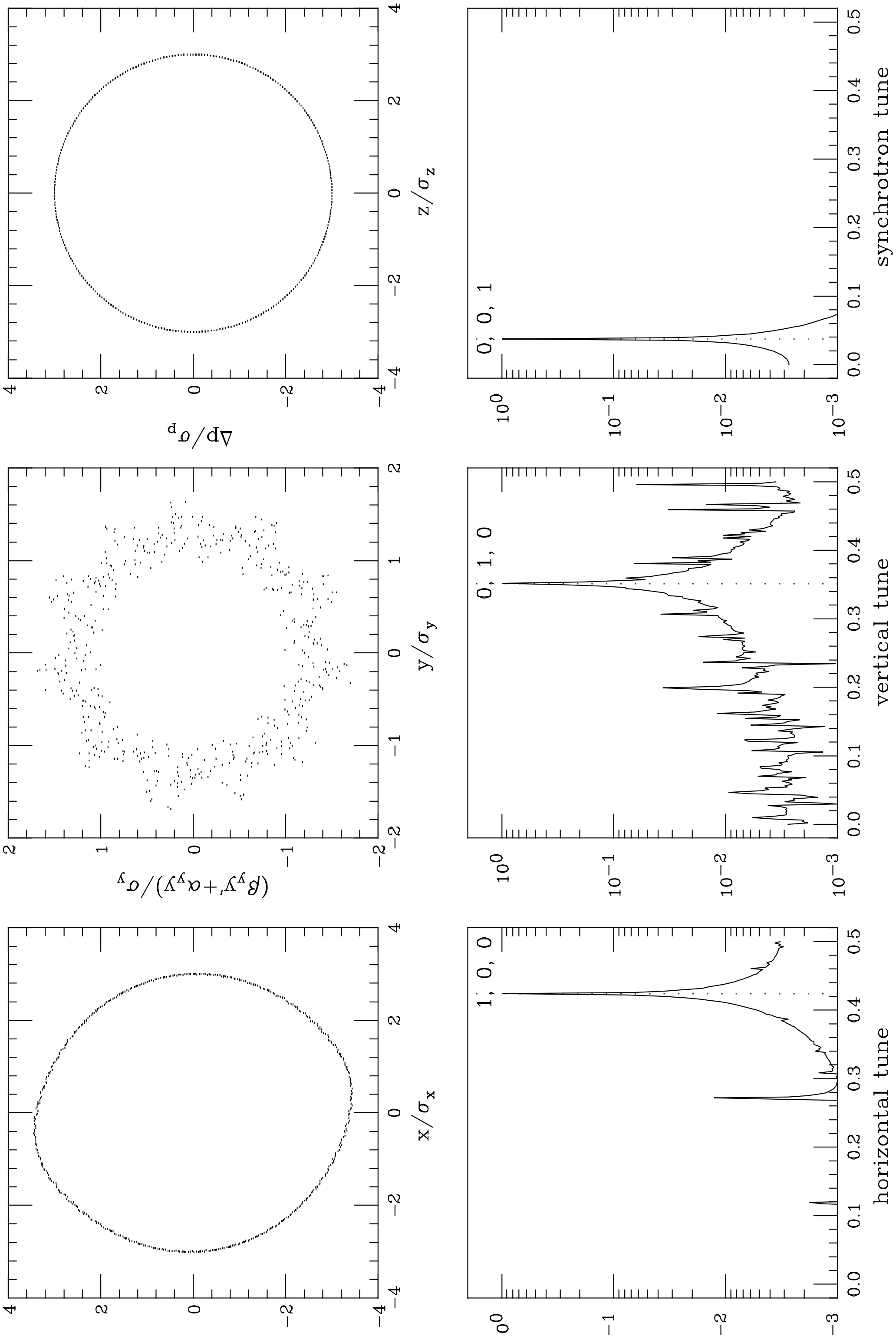

Figure 7: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #4, $w(z)$: alg. #1).

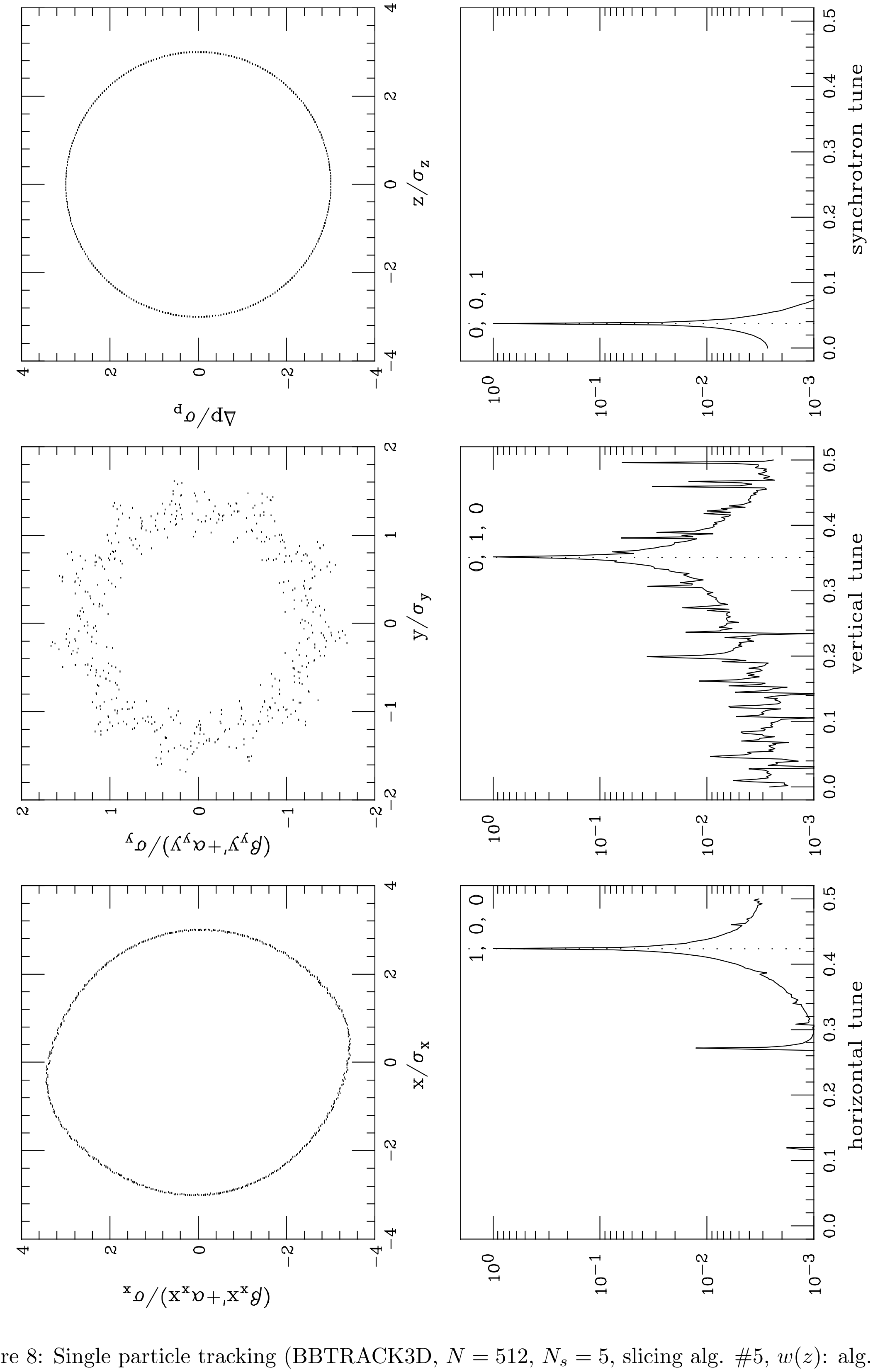

Figure 8: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #5, $w(z)$: alg. #1).

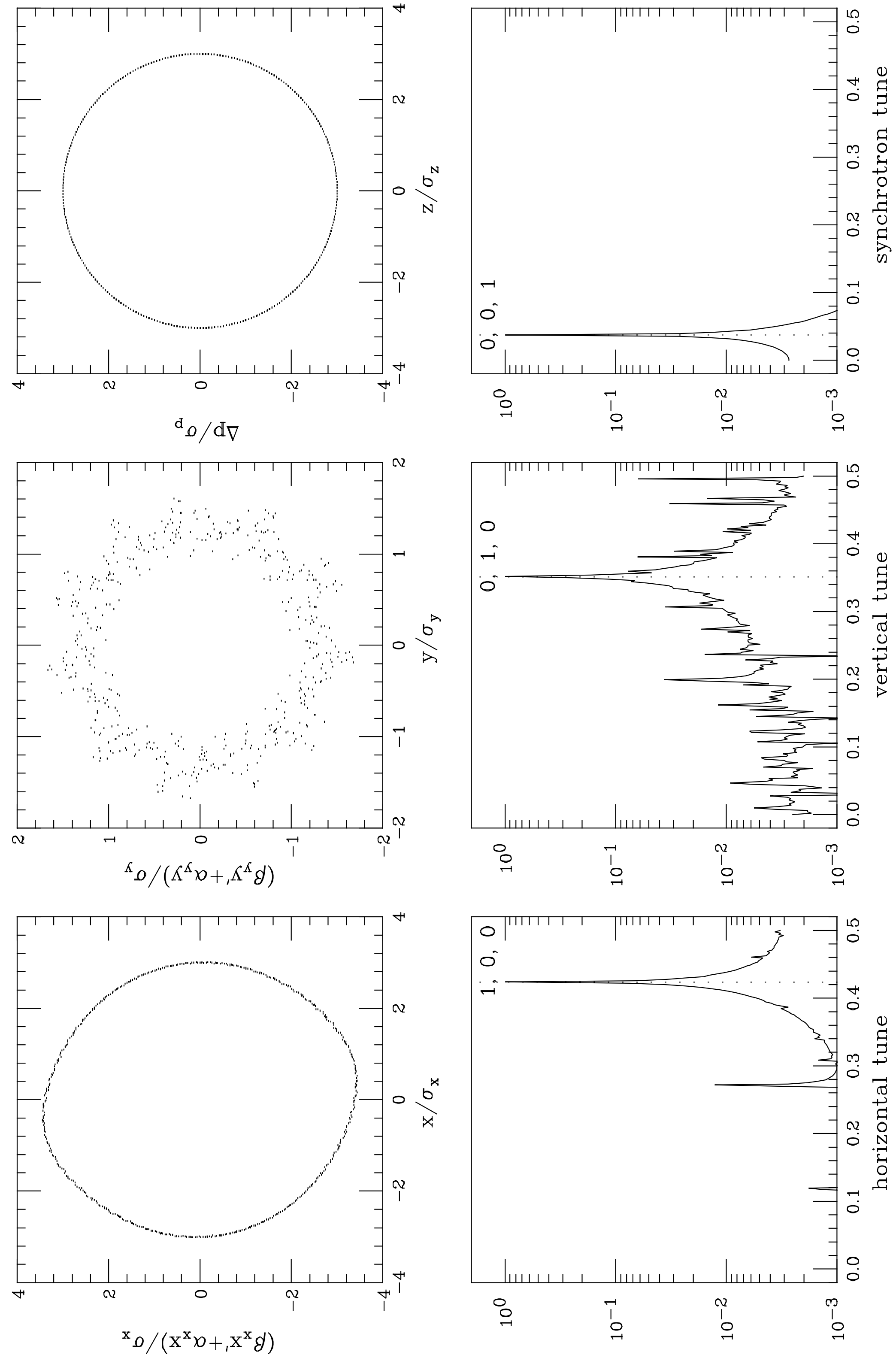

Figure 9: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1). This figure is identical to Fig. 5.

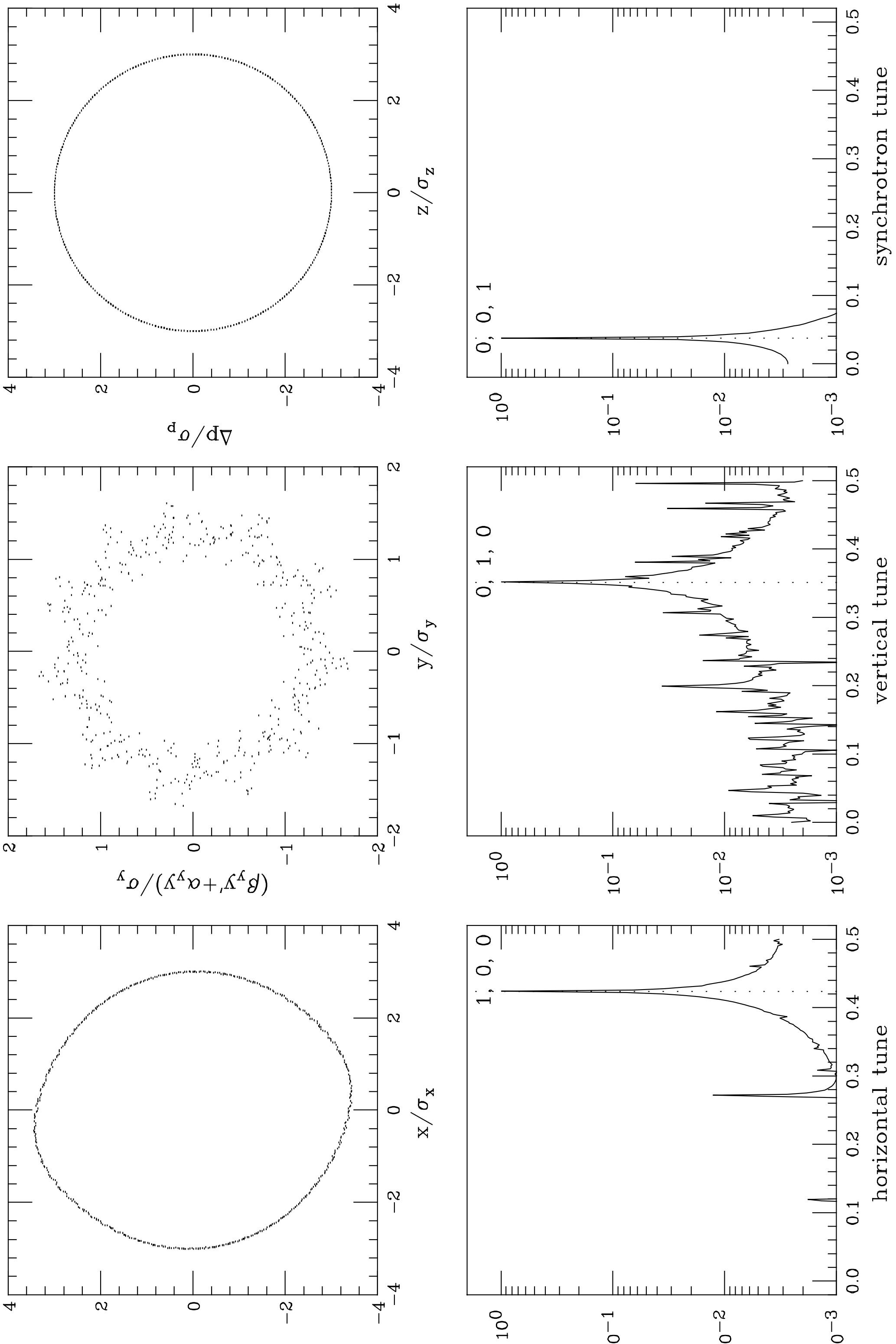

Figure 10: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #2).

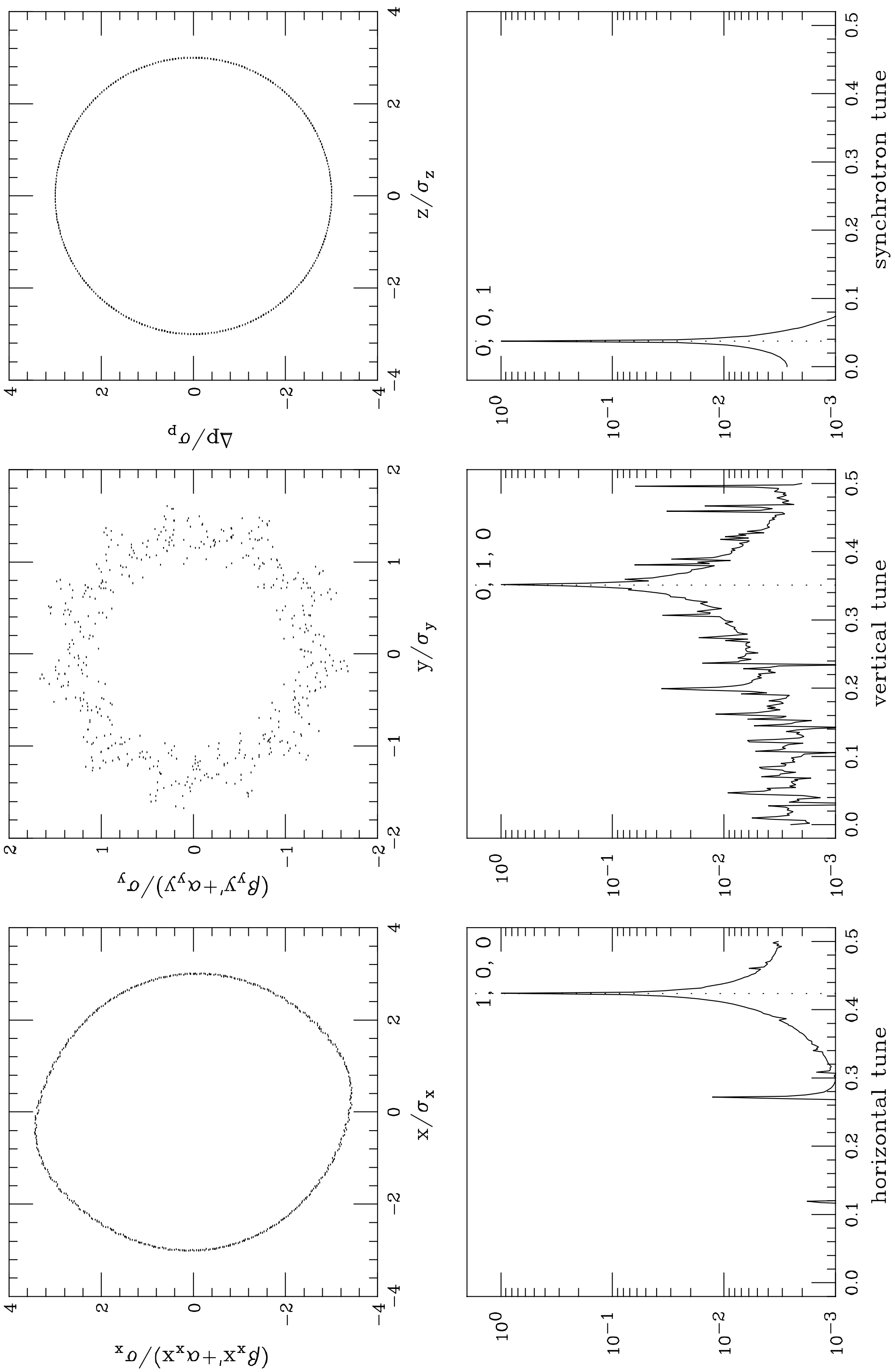

Figure 11: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #3).

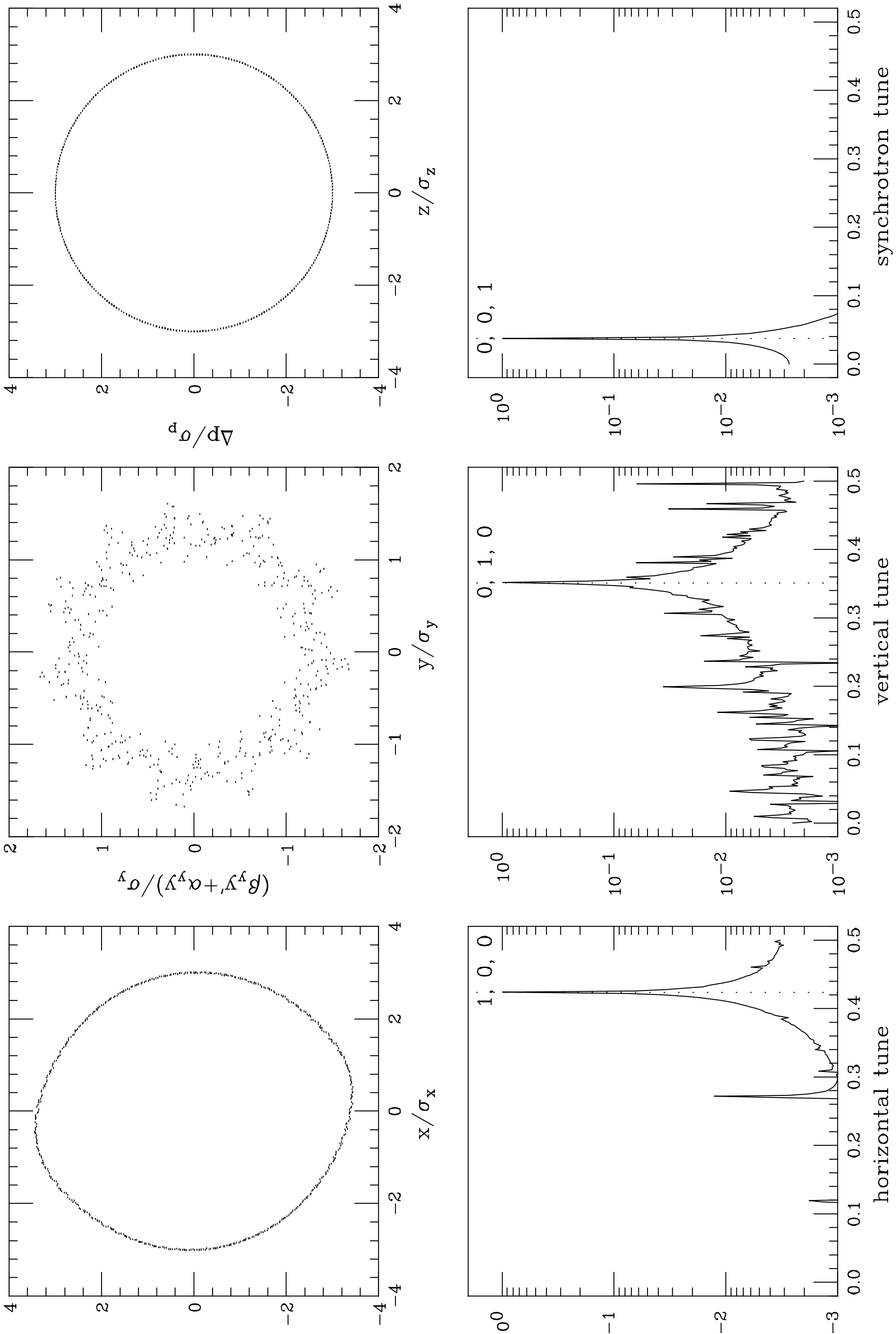

Figure 12: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #4).

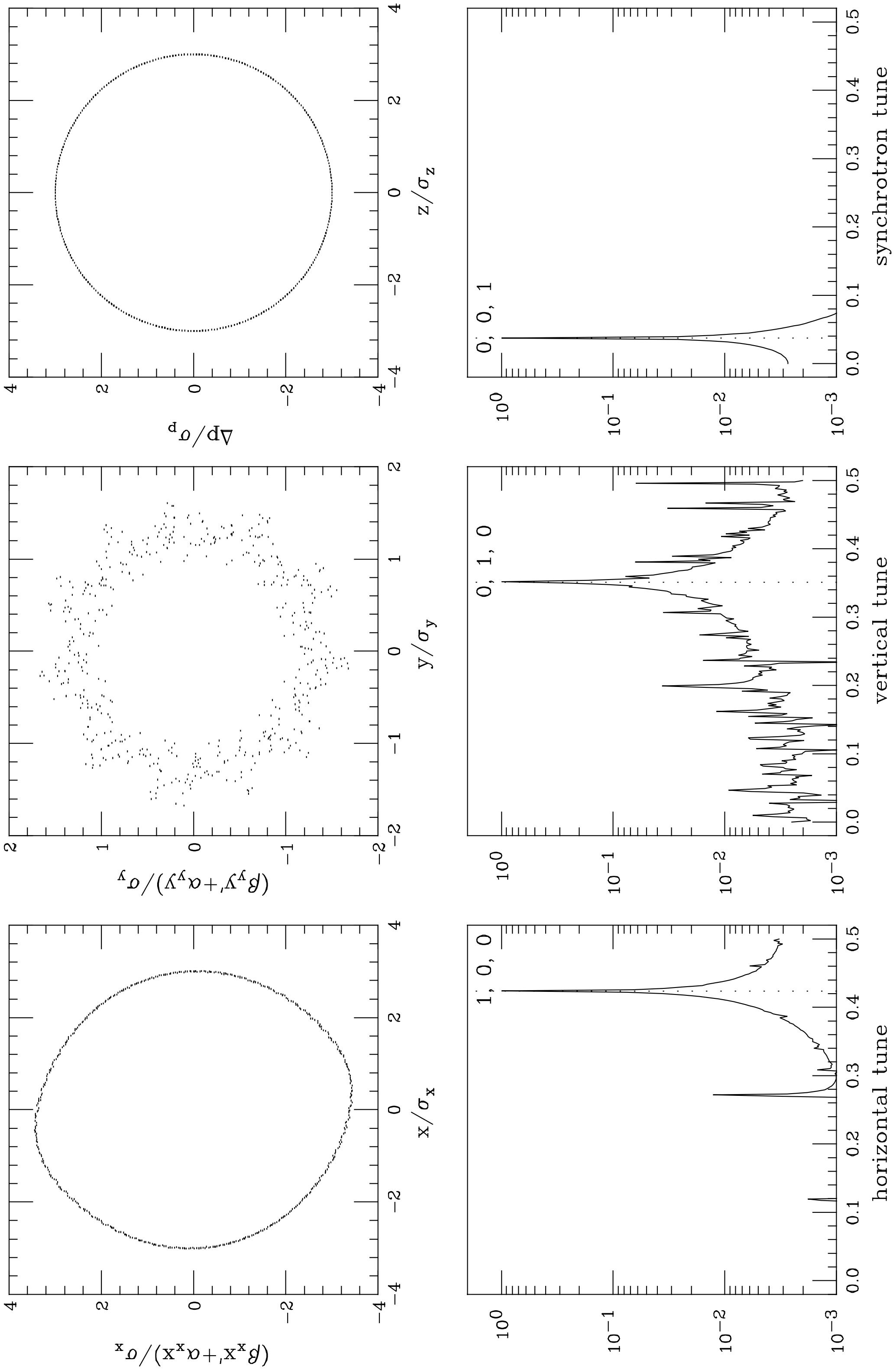

Figure 13: Single particle tracking (BBTRACK3D, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1). This figure is identical to Figs. 5 and 9.

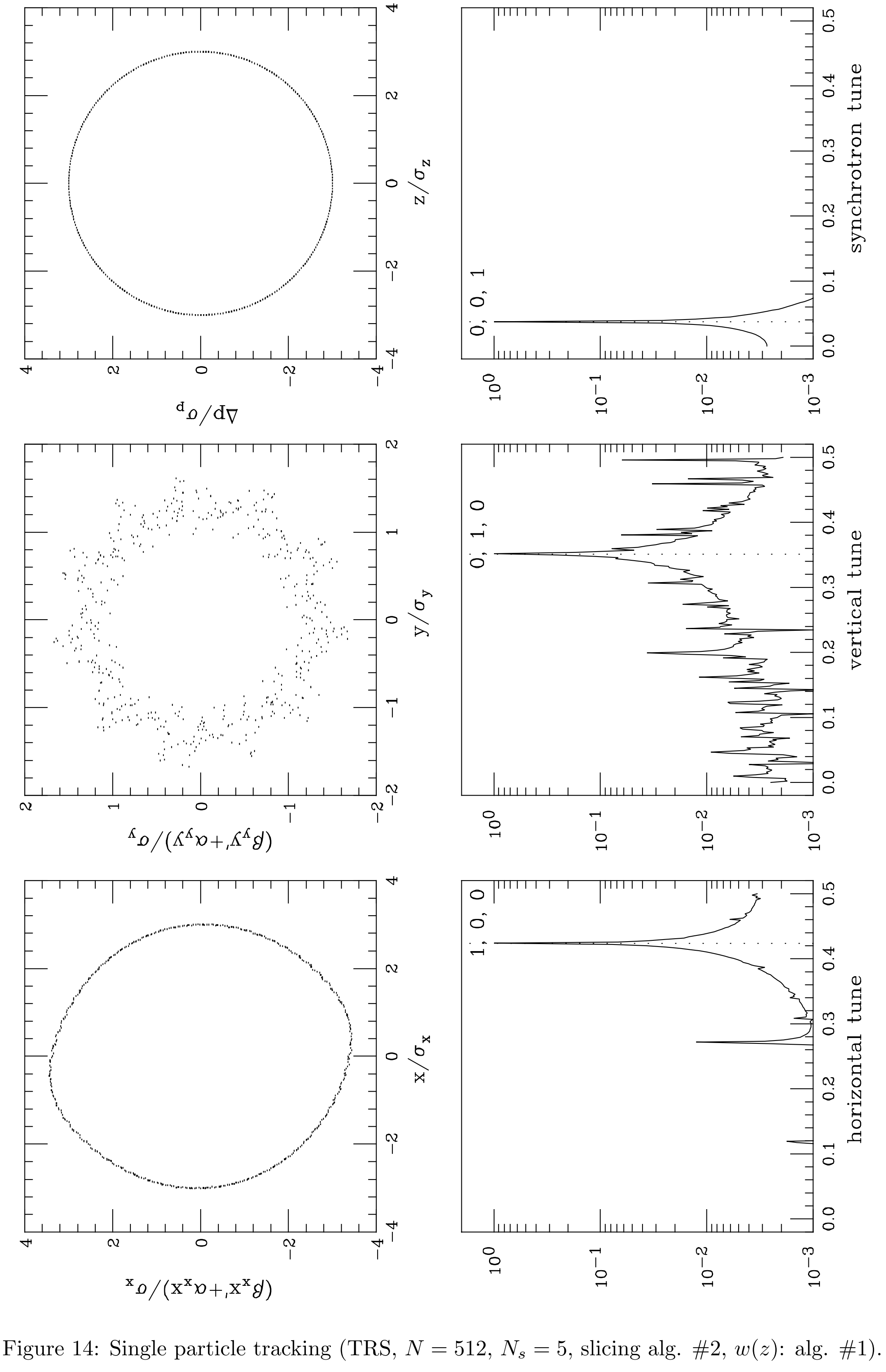

Figure 14: Single particle tracking (TRS, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1).

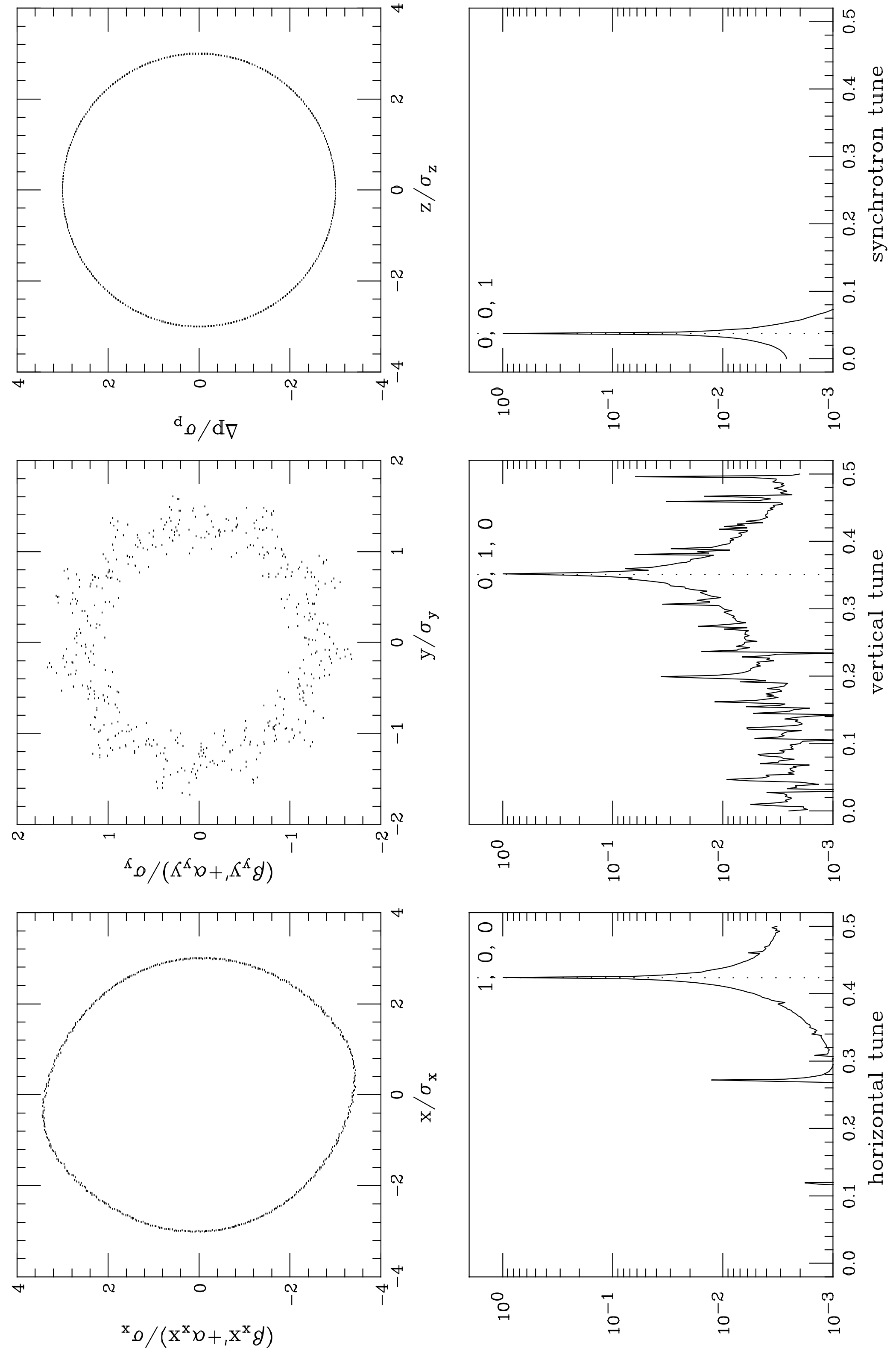

Figure 15: Single particle tracking (LIFETRAC, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1).

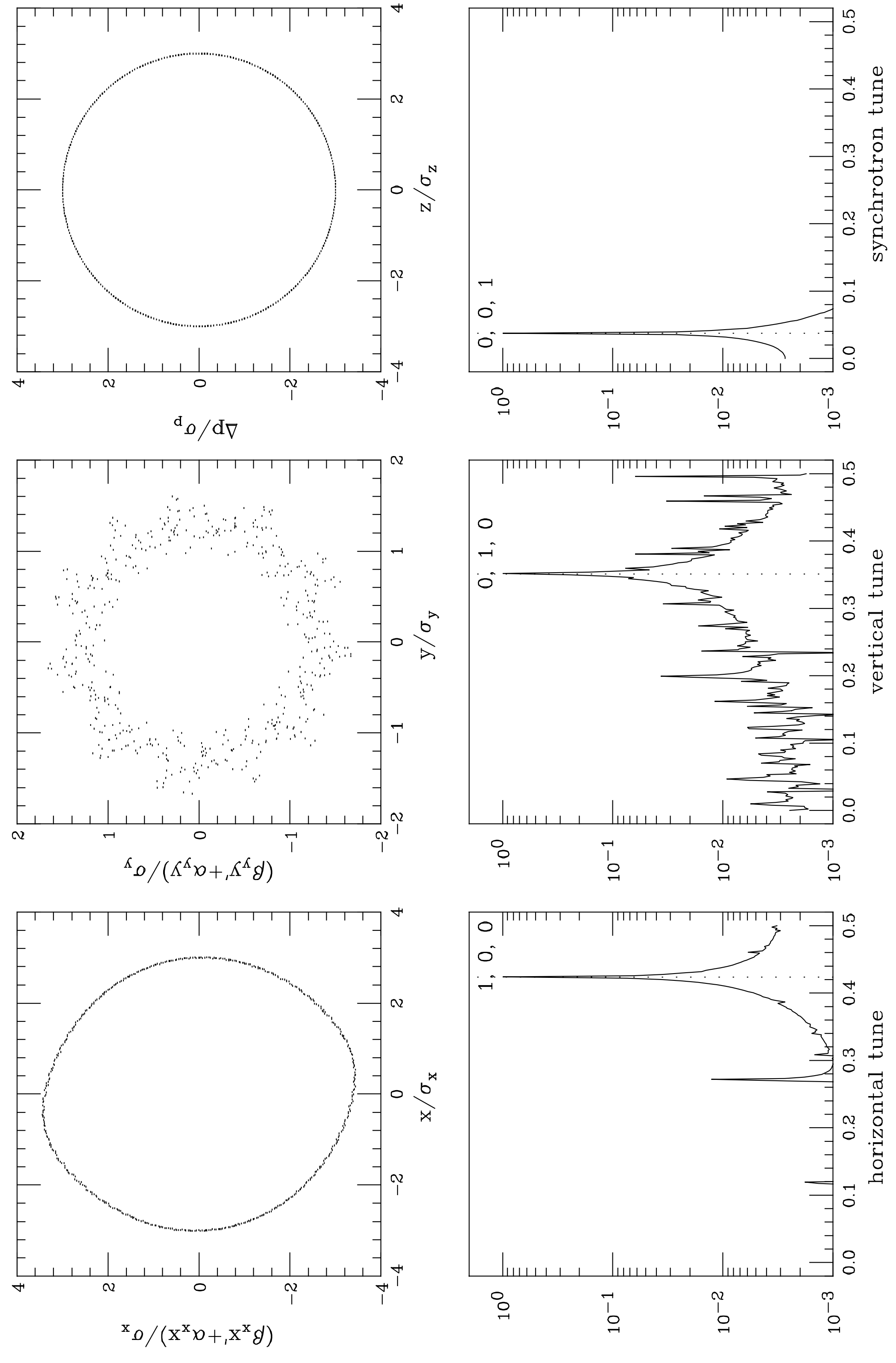

Figure 16: Single particle tracking (TAIL, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #4).

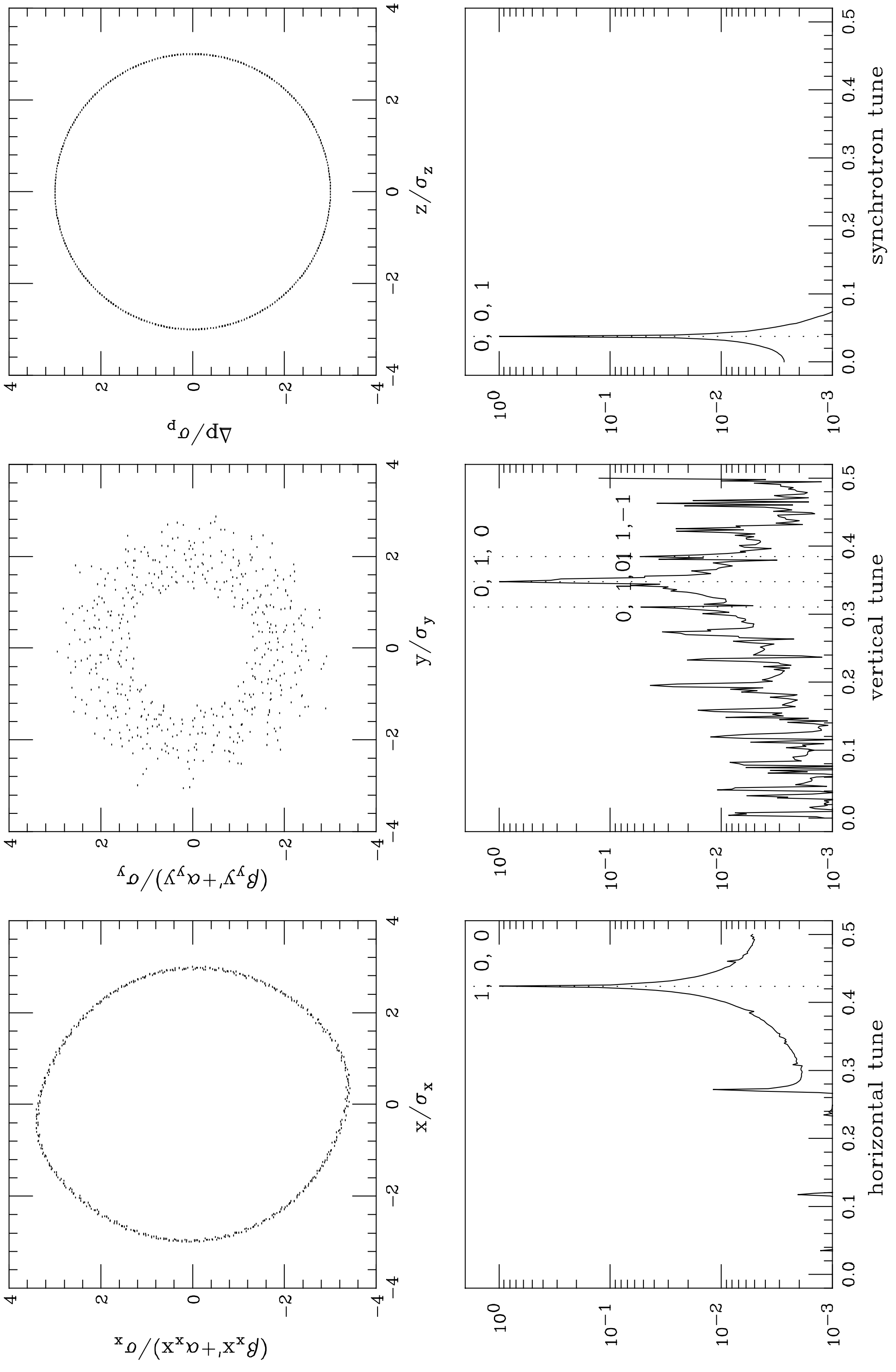

Figure 17: Single particle tracking (TRS with a deliberate error (Eqs. (32)); $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1).

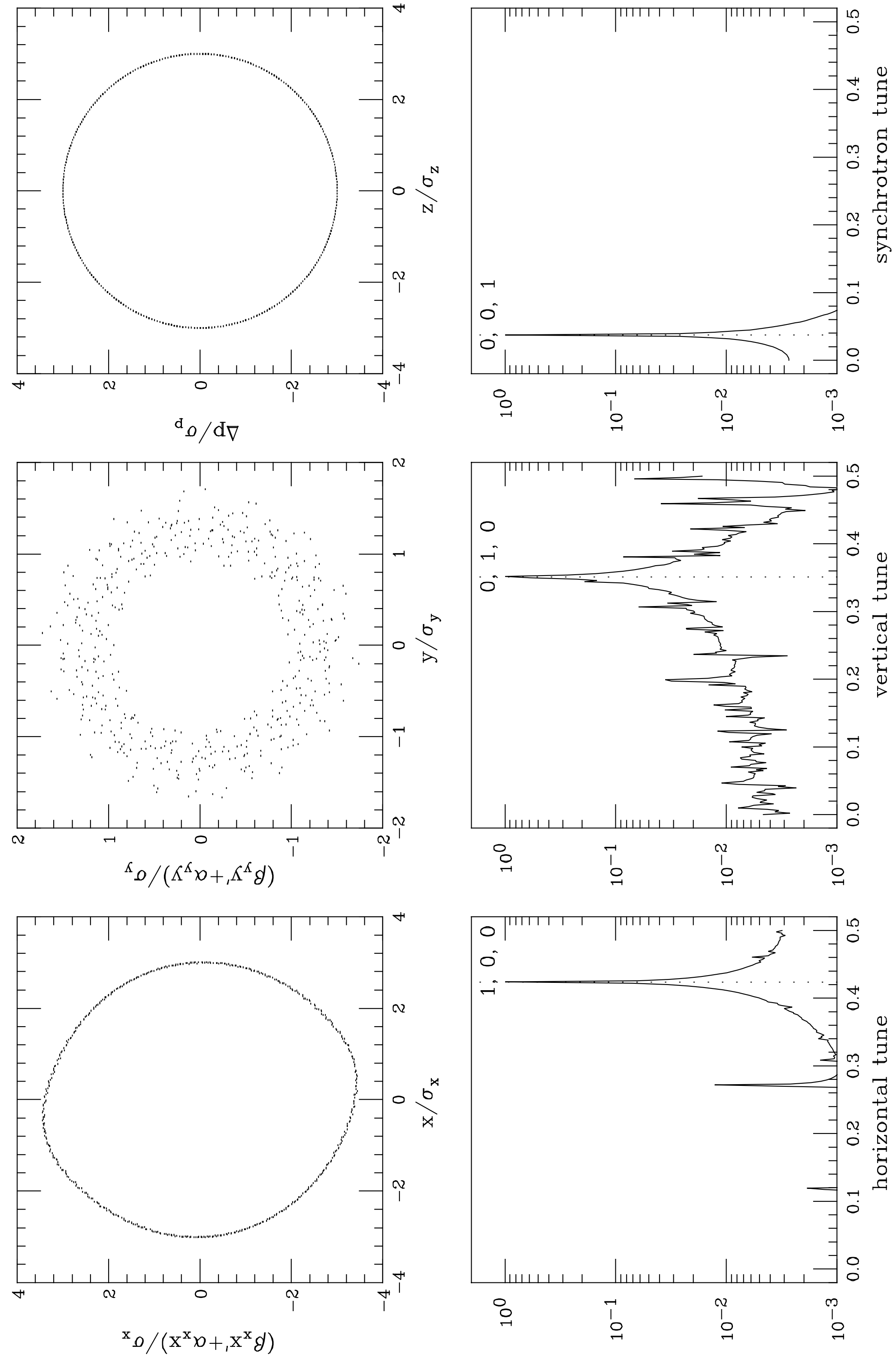

Figure 18: Single particle tracking (TRS with a deliberate error (kicks encountered in reverse sequence), $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1).

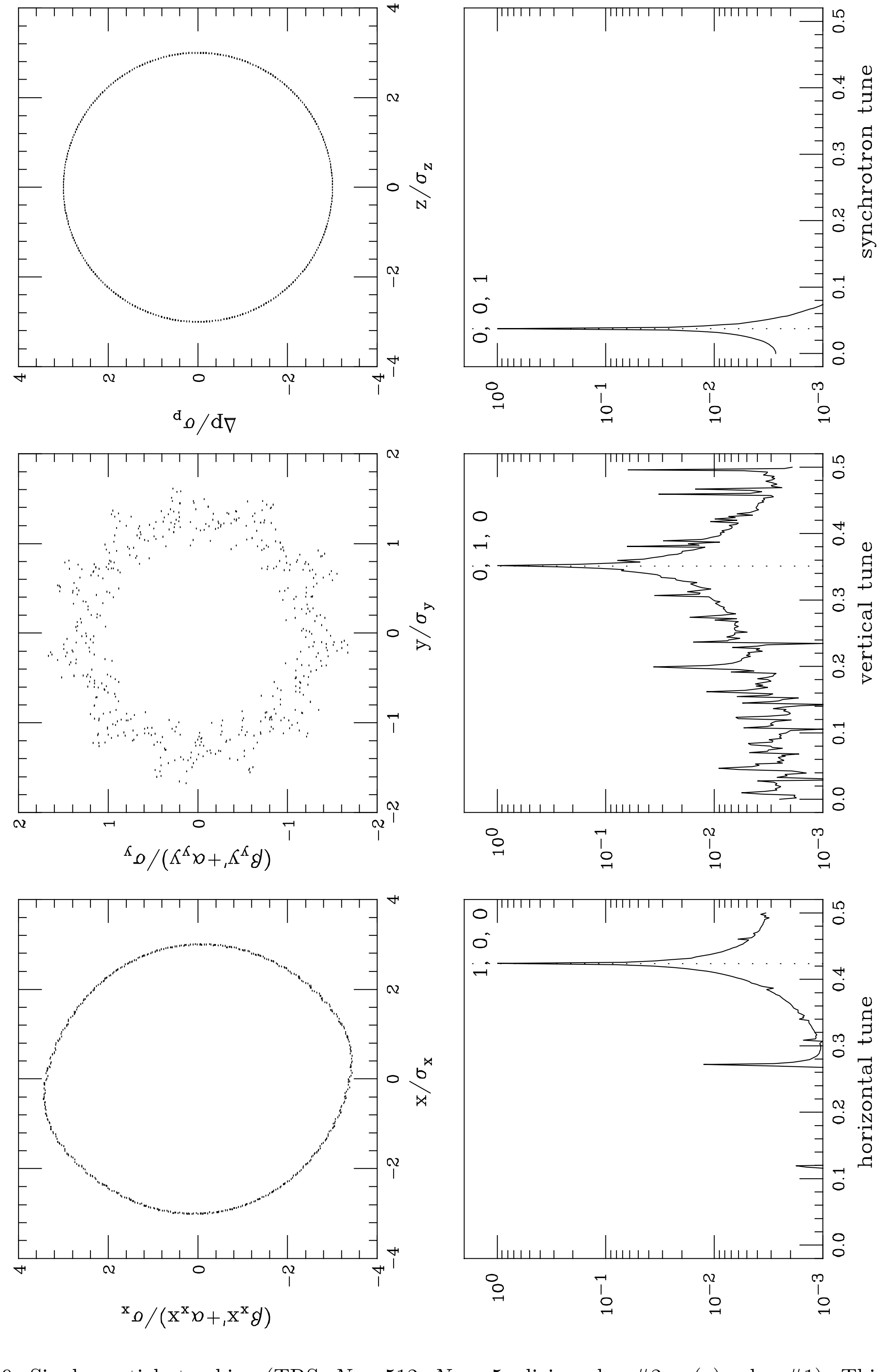

Figure 19: Single particle tracking (TRS, $N = 512$, $N_s = 5$, slicing alg. #2, $w(z)$: alg. #1). This figure is identical to Fig. 14

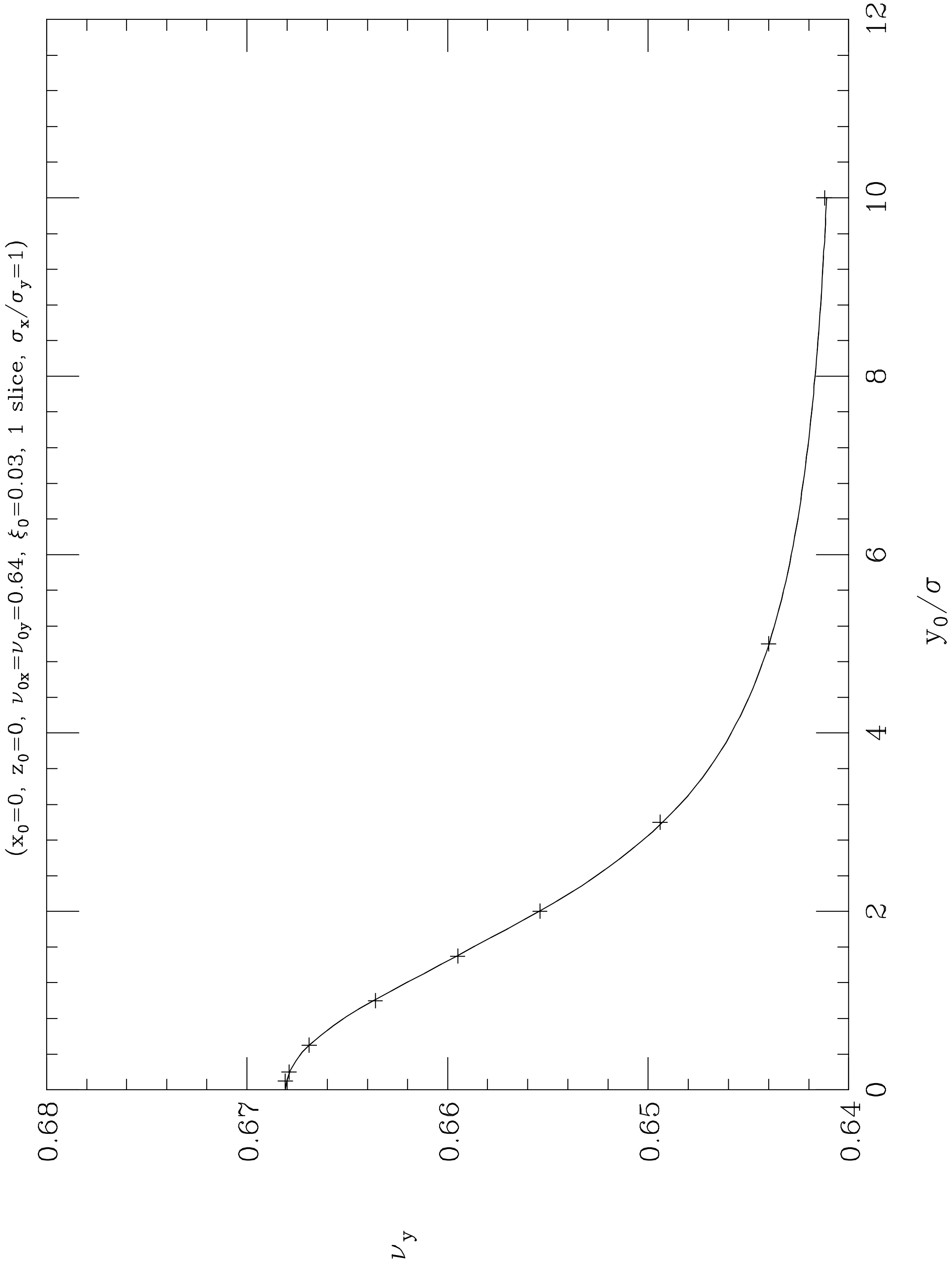

Figure 20: Tune as a function of vertical amplitude for a positron colliding against a round electron beam of rms size σ (see Table 8). Crosses: tracking (TRS, 1 slice, IMSL® library for $w(z)$); solid: analytic (Eqs. (47c) and (39)).

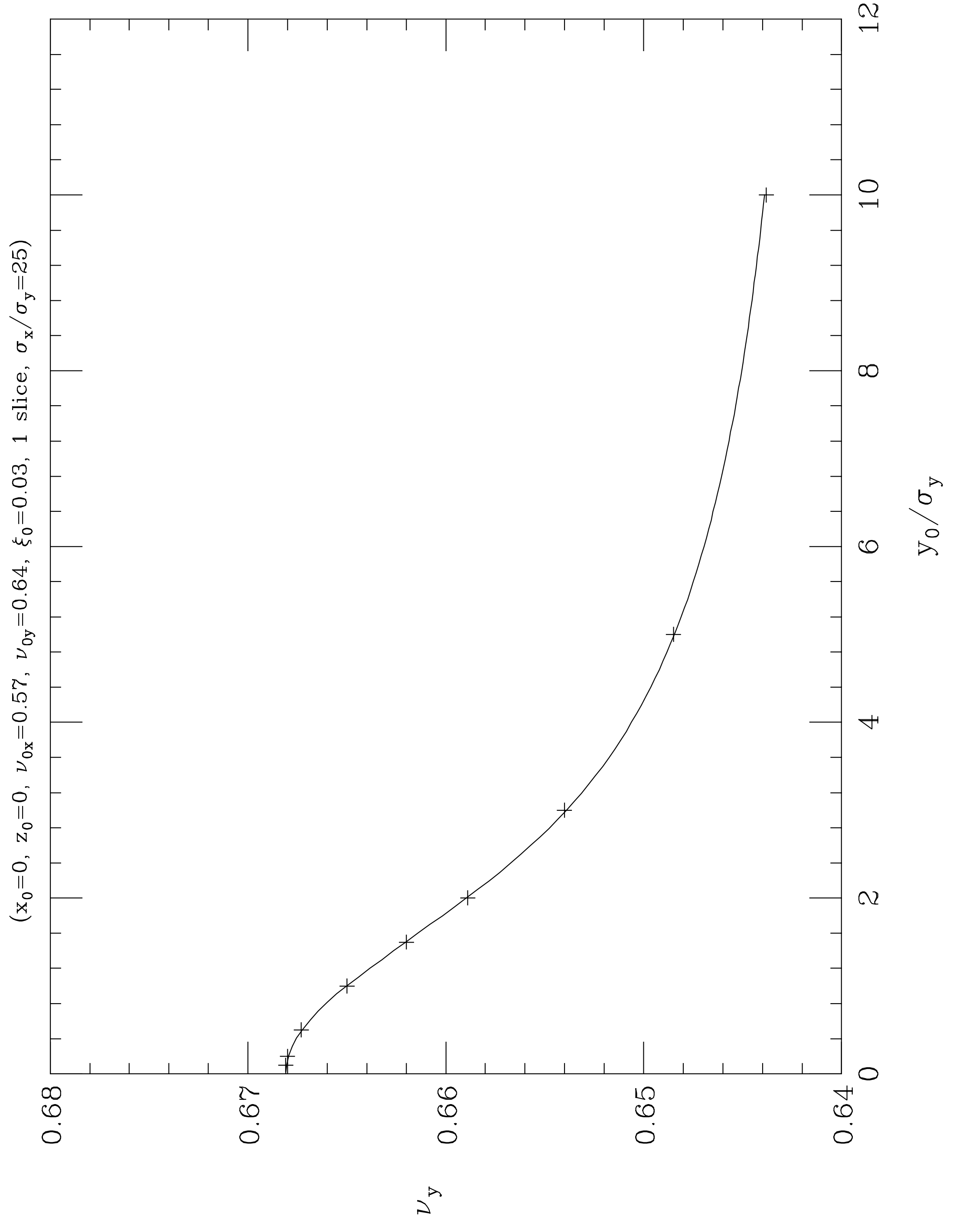

Figure 21: Tune as a function of vertical amplitude for a positron colliding against a flat electron beam of rms sizes σ_x, σ_y for the nominal case (see Table 3). Crosses: tracking (TRS, 1 slice, IMSL® library for $w(z)$); solid: analytic (Eqs. (41d) and (39)).

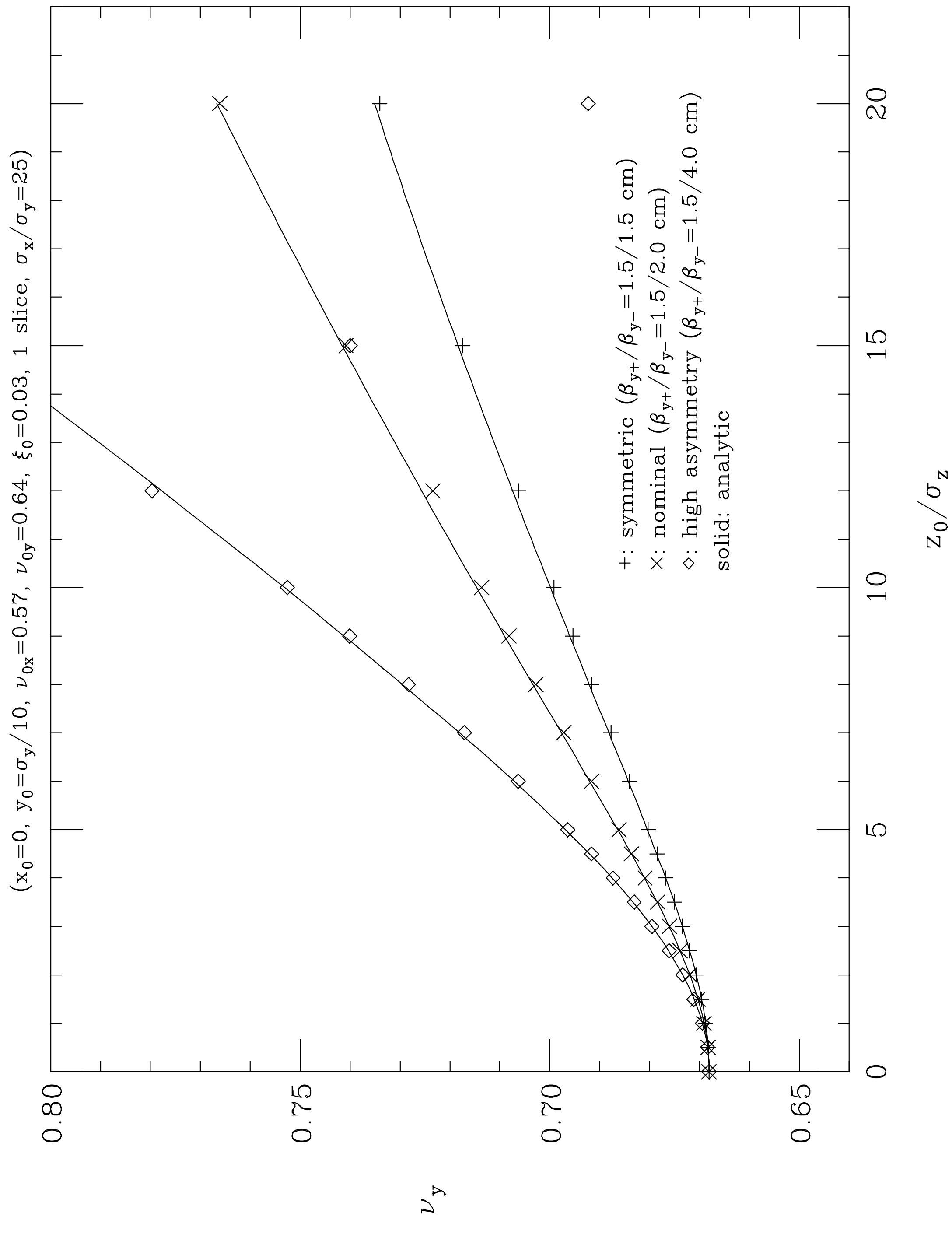

Figure 22: Tune as a function of longitudinal amplitude for a positron colliding against a flat electron beam of rms sizes σ_x, σ_y, σ_z for 3 different values of the beta functions of the electron beam (see Table 9). Crosses: tracking (TRS, 1 slice, IMSL® library for $w(z)$); solid: analytic (Eqs. (52) and (39)).

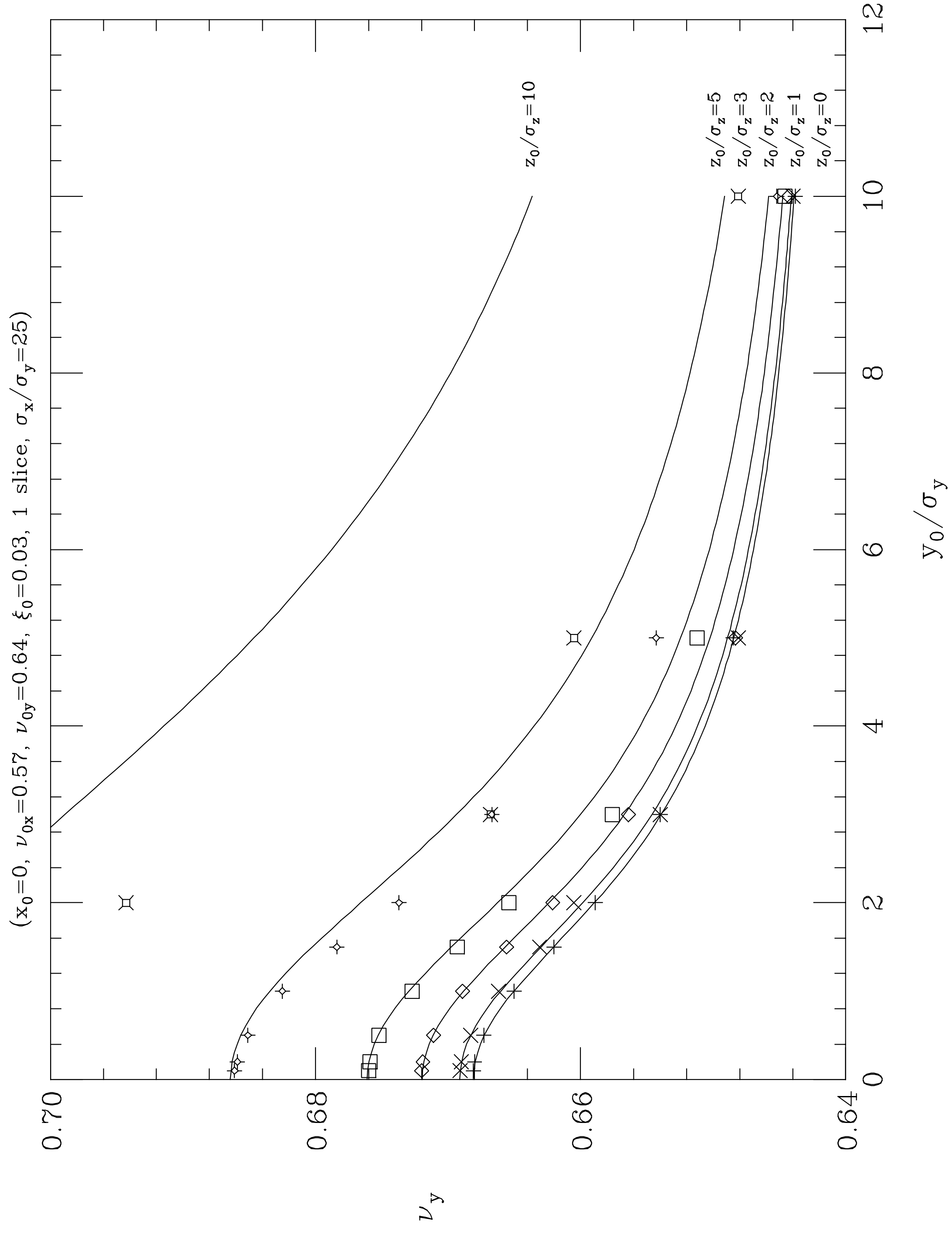

Figure 23: Tune as a function of vertical amplitude for a positron colliding against a flat electron beam of rms sizes σ_x, σ_y, σ_z, for various values of the longitudinal amplitude z_0 of the positron. (see Table 3). Crosses: tracking (TRS, 1 slice, IMSL® library for $w(z)$); solid: analytic (Eqs. (52) and (39)). The lack of agreement between tracking and analytic results for $y_0/\sigma_y \simeq 2$ and $z_0/\sigma_z \geq 1$ is due to the appearance of synchrotron sidebands, clearly seen in Fig. 24.

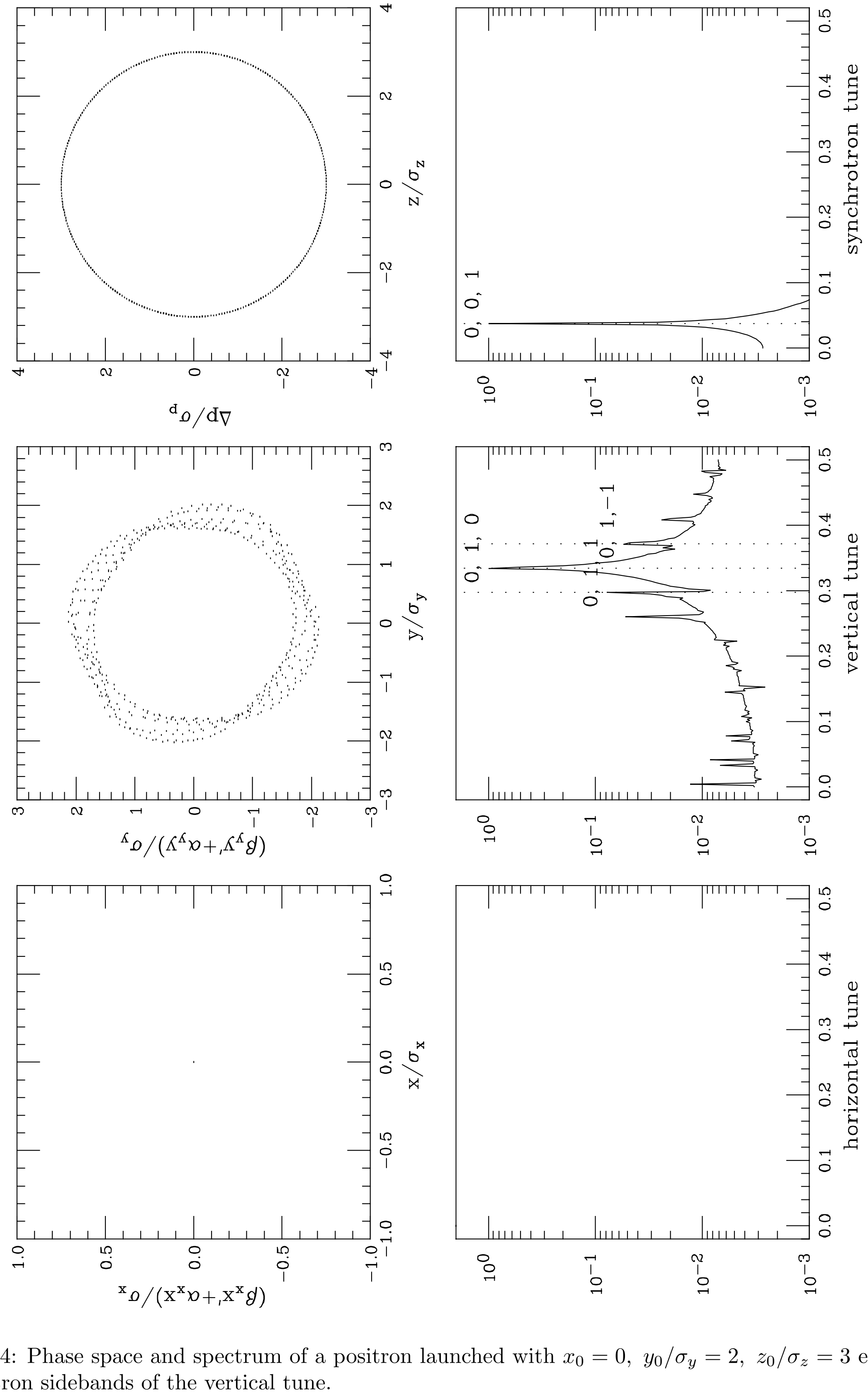

Figure 24: Phase space and spectrum of a positron launched with $x_0 = 0,\ y_0/\sigma_y = 2,\ z_0/\sigma_z = 3$ exhibiting synchrotron sidebands of the vertical tune.

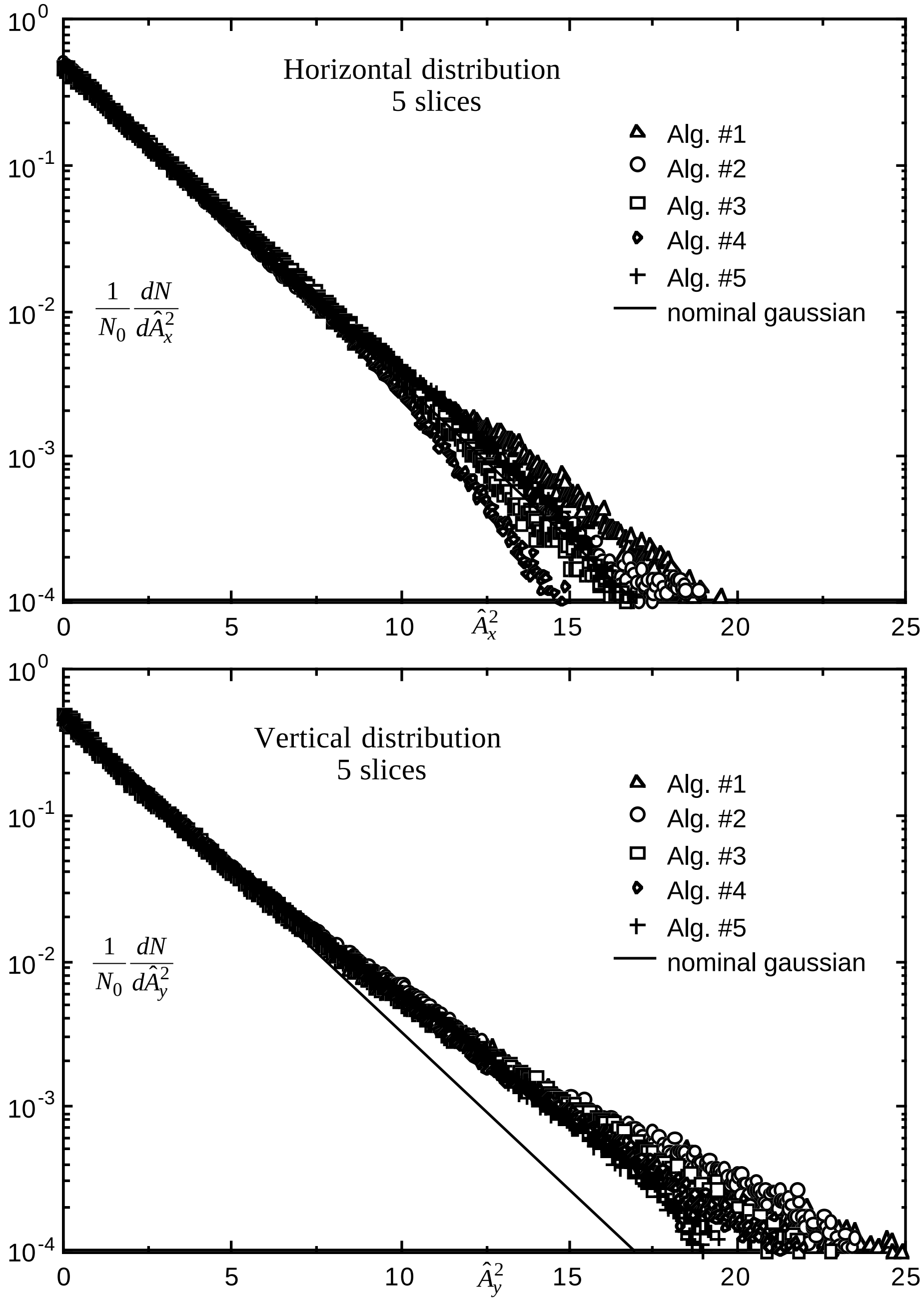

Figure 25: The horizontal and vertical particle distributions for the five slicing algorithms obtained with TRS for $N_s = 5$. The histograms represent 5.12×10^6 particle-turns binned into 500 equal-size bins in $\hat{A}^2$.

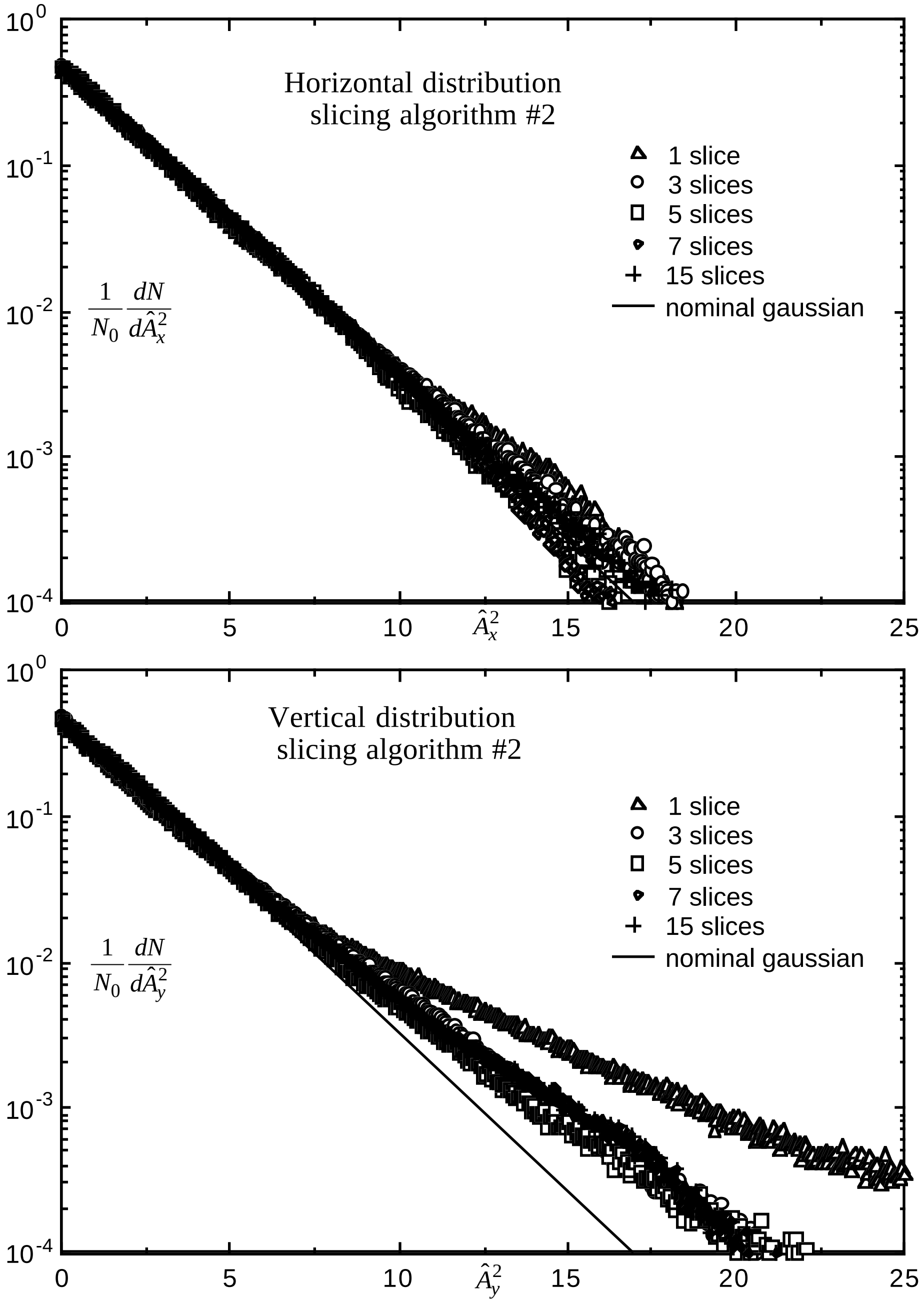

Figure 26: The horizontal and vertical particle distributions obtained with TRS with algorithm #2 by using different number of kicks. The histograms represent 5.12×10^6 particle-turns binned into 500 equal-size bins in $\hat{A}^2$.

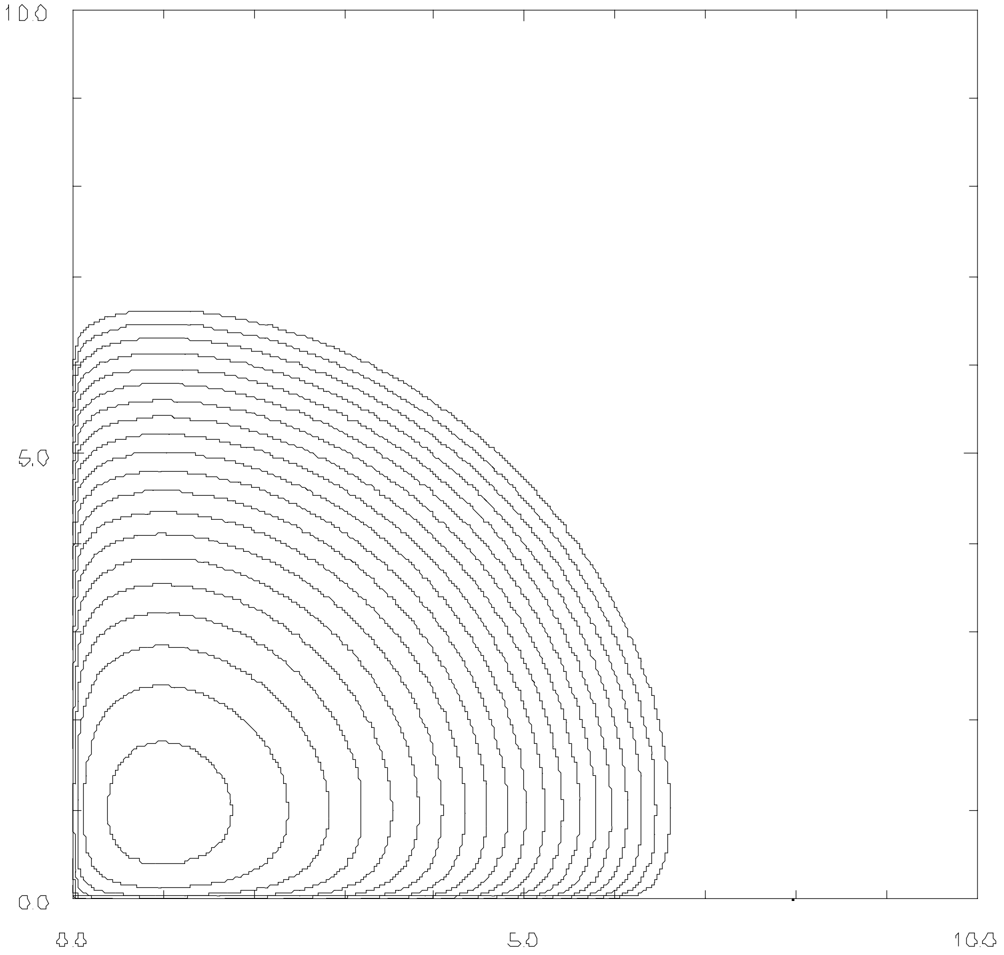

Figure 27: Contour level plot for the nominal gaussian density, Eq. (60). The highest level is at a height $1/\sqrt{e}$ below the peak, and successive contour levels are at a constant ratio e below each other.

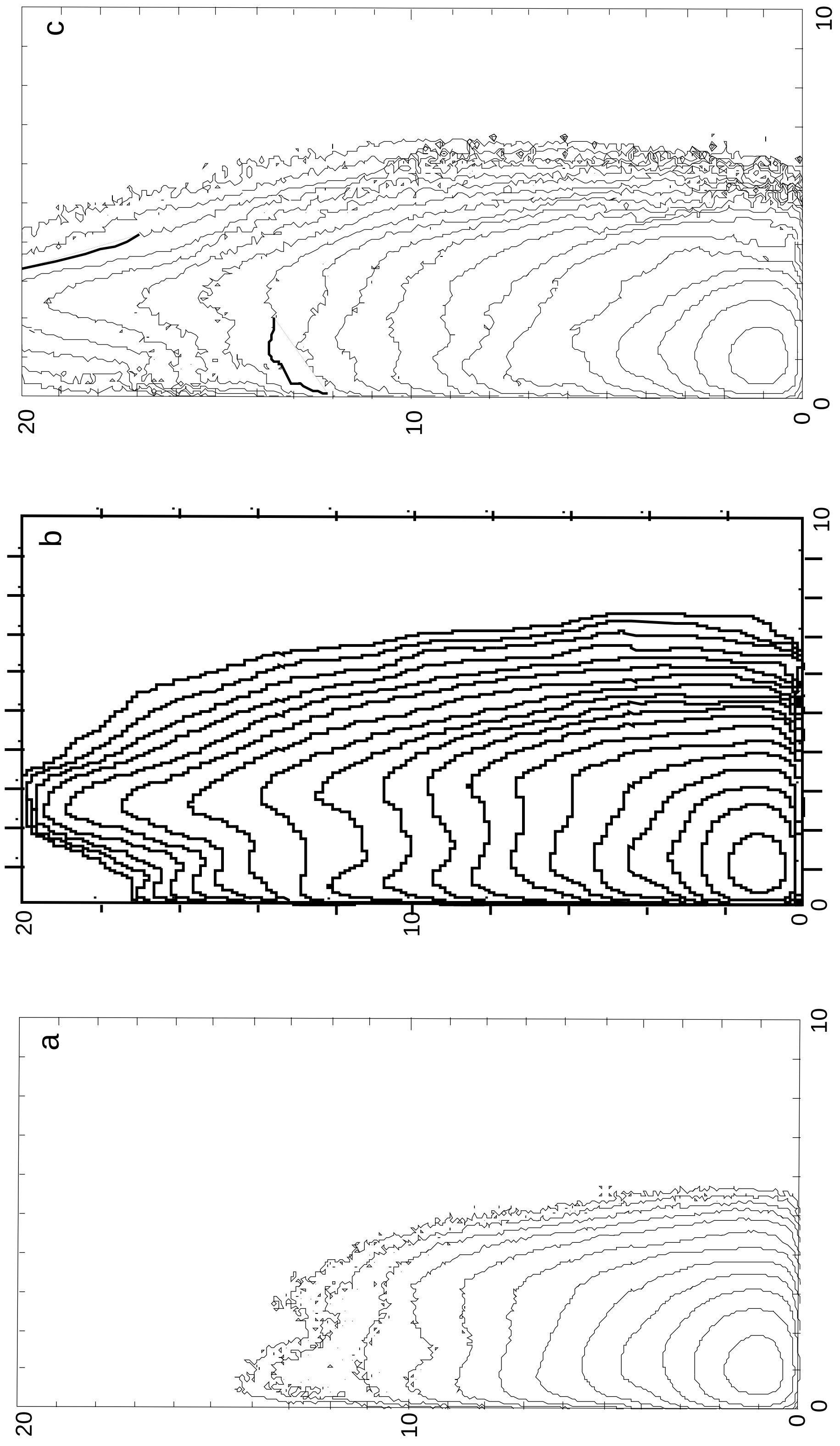

Figure 28: The 2-D amplitude distribution. (a): TRS; (b): LIFETRAC; (c): TAIL. All three cases use 5 kicks; other conditions are listed in Table 12.

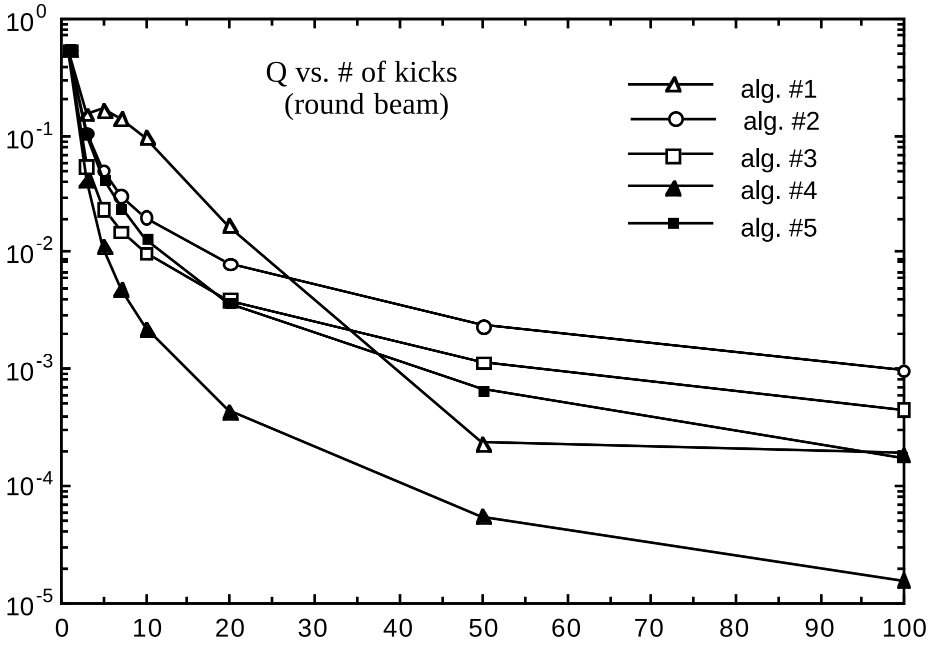

Figure 29: Convergence of the slicing algorithms: Q plotted vs. number of kicks for a round beam; see Table 13.

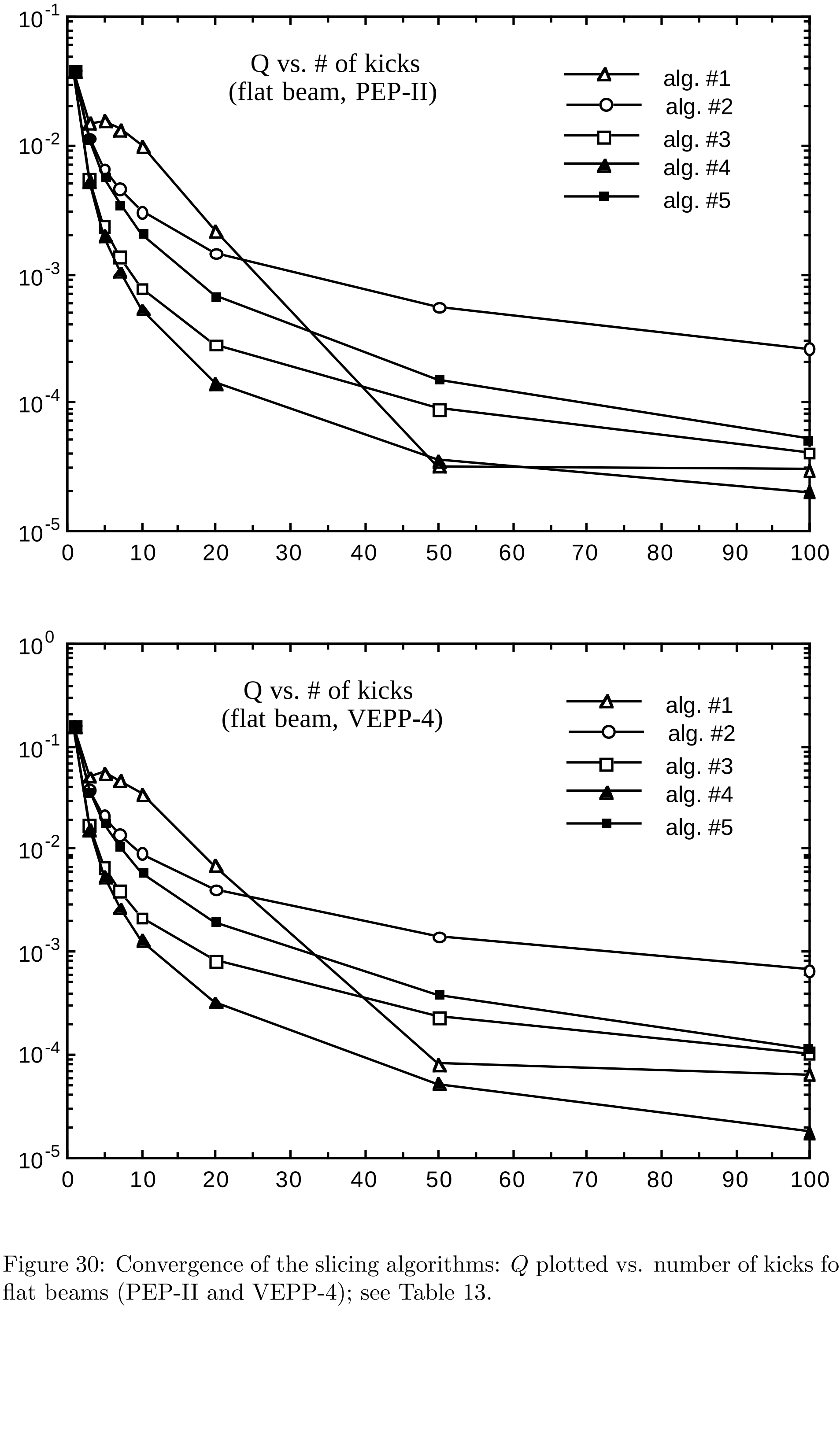

Figure 30: Convergence of the slicing algorithms: Q plotted vs. number of kicks for flat beams (PEP-II and VEPP-4); see Table 13.