Conf-9410229--1

LA-UR- 94-3351

Title:	MOMENT APPROACH TO HIGH-ORDER ACCELERATOR BEAM OPTICS
Author(s):	W. P. Lysenko
Submitted to:	Charged Particle Optics Tsukuba, Japan October 3-6, 1994

Los Alamos
NATIONAL LABORATORY

Form No. 836 R5
ST 2629 10/91

DISCLAIMER

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

THE MOMENT APPROACH TO HIGH-ORDER ACCELERATOR BEAM OPTICS[1]

W. P. Lysenko

AOT-3 MS H808, Los Alamos National Laboratory, Los Alamos, NM 87545, USA

High-current beams must be matched to high order to minimize emittance growth and particle losses. For matching problems, the moment approach, in which we describe the particle beam by the moments of its distribution, is particularly valuable. A variety of analytical results are available for linear motion. The moment approach is also the basis of the 3-D space-charge simulation code BEDLAM, in which the dynamical variables are the moments. Moment simulation codes are particularly useful for computing beams matched to nonlinear systems. This paper outlines what is known about the moment approach, describes work in progress on new space-charge models, and describes further potential applications of and improvements to moment-method simulations.

[1]This work supported by the US Department of Energy, Office of Defense Programs.

INTRODUCTION

For a large class of accelerator problems, we are interested in properties of the beam as a whole rather than in the single-particle motion. For example, when we are putting beam on a target, we usually do not care what the individual particles are doing. We just want the spot to be a certain size. In situations like these, it is advantageous to use the moment approach, in which we describe the beam by the moments of its phase-space distribution. In our approach, emphasis is more on the beam than on the beamline. An advantage of this is that space charge, which, of course, depends on the beam distribution, can be included in a natural way.

When we deal with high-current beams, high-order effects (nonlinearities in the forces and finer details of the beam distribution) become important. Nonlinearities arise both from the focusing elements and from particle interactions (we assume particle interactions are in the electrostatic Vlasov regime, in which the space-charge forces depend only on the distribution function).

An important problem is the reduction of emittance growth and beam loss, which we believe can be achieved by matching the beam to to the beamline (which may contain aberrations) to as high an order as possible. This differs from conventional approaches that try to eliminate aberrations to as high an order as possible. We believe our approach makes more sense in the high-current regime. Aberration reduction will have to involve beam distribution manipulation anyway, since space-charge forces are contributing significantly to the motion.

MOMENT DESCRIPTION

Consider a bunched beam in a rf linear accelerator or in some beamline downstream

of such an accelerator. The bunches are roughly ellipsoidal. We use a coordinate system fixed at the center of a bunch as shown in fig. 1. As the beam bunch travels down the machine, the forces seen by the particles in the bunch vary in time. The problem is to determine how $f(\vec{x},\vec{p},t)$, the six-dimensional phase-space distribution function that describes this bunch, evolves.

We describe the distribution f by its moments, which are just averages of monomials in the phase-space variables. For example, the moment $<x^2>$ is defined by

$$<x^2> = \int_{-\infty}^{\infty} d^3p \int_{-\infty}^{\infty} d^3x \, f(\vec{x},\vec{p},t)\, x^2 \tag{1}$$

$$= \frac{1}{N}\sum_{i=1}^{N} x_i^2. \tag{2}$$

The discrete case, eq. (2), is useful if we want to compute moments in a particle code. It is just a sum over particles. The moments have the desirable property that they are closely related to to observables such as beam positions (first moments) and sizes (second moments).

MOMENT INVARIANTS

The usual rms emittance is defined for one-dimensional (1-D) motion[2]. It is a function of the second moments. In our notation, the rms emittance in the x-direction is

$$\epsilon_2 = (I_{22})^{1/2} = (<x^2><p_x^2> - <xp_x>^2)^{1/2}. \tag{3}$$

There are similar definitions for the x and y directions. The rms emittances are like areas in the 2-D projections of the 6-D phase space. For example, in the rotated coordinate

[2] Unless we explicitly say n-D phase space, by n-D we mean n degrees of freedom, which corresponds to $2n$-D phase space.

system in which the correlation $<xp_x>$ is zero, ϵ_2 is the the product of the rms values of x and p_x.

The importance of the rms emittance comes from the following dynamical property: it is conserved for linear, uncoupled (1-D) motion. The rms emittances can change either if there are couplings between the the degrees of freedom or if there are nonlinear forces. Even *linear* coupling, as caused by rotated quadrupoles, for example, can cause the rms emittances to grow.

The idea of rms emittance can be extended to 2-D and 3-D. We have found that the following function of second moments is invariant for linear motion in 2-D.

$$\begin{aligned} I_{22} = & <x^2><p_x^2> - <xp_x>^2 \\ & + <y^2><p_y^2> - <yp_y>^2 \\ & + 2<xy><p_xp_y> - 2<xp_y><yp_x> \end{aligned} \tag{4}$$

Notice that this invariant consists of the sums of the squares of the rms emittances in the two directions, together with some cross terms. This quantity is preserved even if the forces are coupling the x and y directions. A similar-looking 3-D version of this invariant also exists. For 2-D and 3-D there is an invariant I_{2222} that is of degree four in the second moments and for 3-D there is an invariant I_{222222} that is of degree six in the second moments. There is an efficient algorithm for computing the numerical values of such invariants, given the second moments[1]. These invariants are useful as diagnostics in simulation codes. If we see see that these second-moment invariants are not changing, then we know that any observed emittance growth is being caused by couplings rather than by nonlinearities.

Invariants for higher moments also exist. The following function is of degree two in

the fourth moments. It is invariant for linear motion in 1-D.

$$I_{44} = <x^4><p_x^4> - 4<x^3p_x><xp_x^3> + 3<x^2p_x^2>^2 \tag{5}$$

Invariants of higher moments also exist. See ref. [2]–[4] for more information.

For 1-D motion, we can define n-th order emittances as follows. For ϵ_2 and ϵ_4 we have

$$\begin{aligned} \epsilon_2 &= (I_{22})^{1/2} = (<x^2><p_x^2> - <xp_x>^2)^{1/2} \\ \epsilon_4 &= (I_{44})^{1/4} = (<x^4><p_x^4> - 4<x^3p_x><xp_x^3> + 3<x^2p_x^2>^2)^{1/4} \end{aligned} \tag{6}$$

In general we define

$$\epsilon_n = (I_{nn})^{1/n}, \tag{7}$$

where I_{nn} is the moment invariant of degree two in the n-th moments for 1-D. All the ϵ_n are preserved for linear motion in 1-D, even for mismatched beams, and have dimensions of areas in phase space.

It is well known that a matching section in a beamline composed of linear elements like magnetic quadrupoles can be used change the Courant-Snyder parameters α and β of a beam to whatever values are desired but cannot change its rms emittance. In a similar way, the ϵ_n and other invariants of higher moments describe the higher-order features of beams that cannot be changed by linear matching sections.

Table 2 shows the values of ϵ_n for various orders n for two distributions: a uniformly-filled disk ($x^2 + p_x^2 \leq 1$) and a hollow beam in which all particles are on the circle $x^2 + p_x^2 = 1$. The value for ϵ_2 for the uniform beam is the familiar result that the rms emittance is one-fourth of the "total emittance." We see how the higher-order emittances are more sensitive to the outer portions of the beam (halo region). For high n, the interior of the beam does not matter; ϵ_n is approximately measuring the

outermost area occupied by the beam. Because of this property, the higher emittances should be approximately invariant for nonlinear forces because phase-space volume is preserved even for nonlinear forces. For 3-D, there are other higher-order invariants and these will correspond to the other Poincaré invariants.

MATCHING

Most discussions of higher-order particle optics involve aberrations, which are often eliminated by introducing symmetries into the beamline. This approach is essential in imaging systems, which must transmit any beam well (where we are not free to change the input beam in order to get the desired output beam). However, we will discuss another approach, which is appropriate to a large class of particle-beam applications. This is the idea of matching beams to the beamline structure, which may have aberrations. Here, we are introducing symmetries into the beam rather than into the beamline in order to solve our problem, usually the minimization of emittance growth and particle loss. Since we are concerned with the evolution of the distribution rather than the single-particle motion, the moment approach is particularly valuable.

What is matching?

To understand matching, first consider a time-independent, linear force in 1-D. The single-particle motion is described by a harmonic-oscillator Hamiltonian.

$$H = \frac{1}{2m} p^2 + \frac{m\nu^2}{2} x^2 \tag{8}$$

Solutions are of the form

$$x = A\cos(\nu t + a) \tag{9}$$

and single-particle invariants of the motion are ellipses

$$\gamma x^2 + \beta p^2 = \text{const.}, \tag{10}$$

as shown in fig. 2. There is only one parameter, the focusing strength, which is described by the betatron frequency ν.

In general, a matched beam is defined as one whose distribution function is a function of the single-particle invariants. In the present case, we have

$$f(x, p_x) = F(\gamma x^2 + \beta p_x^2), \tag{11}$$

where F is an arbitrary function. The phase-space density must be constant on any ellipse but the density can vary from ellipse to ellipse. An important property of matched beams is that the matched beam size decreases as the focusing strength increases.

We can scale the variables x and p_x to make the invariant ellipses be circles so the evolution of any distribution is just a rotation. Fig. 3 shows this situation. The mismatched distribution shown (a disk with a piece cut out) rotates in time. Obviously, the centroid and all other moments of this distribution are going to oscillate at the betatron frequency. If the distribution were matched (no piece cut out), rotations would leave the distribution unchanged and the distribution (and thus all its moments) would be time independent, even though the individual particle motions are oscillating at the frequency ν.

Now consider a realistic focusing situation, an alternating-gradient focusing lattice consisting of alternating focusing and defocusing quadrupoles as show schematically in fig. 4. Suppose it takes time T to travel one period in this lattice. The focusing forces seen by the particles are now periodic in time

$$H = \frac{1}{2m} p_x^2 + \frac{m\nu^2}{2} K(t)\, x^2. \tag{12}$$

The function $K(t)$, which has period T, describes the focusing strength of the various

beamline elements. The solutions are

$$x = \sqrt{\beta(t)} A \cos(\nu t + a(t)), \tag{13}$$

where the beta function $\beta(t)$ and the phase $a(t)$ are periodic functions with period T. Figure 6 shows a typical solution. We see that there two frequencies present: the slow betatron frequency ν (the figure shows one betatron oscillation) and the fast frequency corresponding to the period T. In this example, we have a phase advance per period of 60°, which means that the betatron period is six focusing periods long.

The single-particle invariants are ellipses

$$\gamma x^2 + 2\alpha x p_x + \beta p_x^2 = \text{const.} \tag{14}$$

whose coefficients α, β, and γ are periodic functions of time. These invariant curves are called the Courant-Snyder invariants. The β in the the ellipse of eq. (14) is the same function appearing in eq. (13). The periodic functions α and γ are simply related to the β function. This situation is shown in fig. 5. The ellipses at time t and time $t + T$ are the same even though the location of any given point on the ellipse moves to a new location (as shown by the heavy dot in the figure).

Distributions matched to periodic, linear beamlines are functions of the invariant ellipses

$$f(x, p_x, t) = F(\gamma x^2 + 2\alpha x p_x + \beta p_x^2). \tag{15}$$

These distributions are periodic in time. We see that matched distributions contain only frequencies present in the Hamiltonian. The betatron frequencies present in the single-particle motion are not present because of the symmetry we have imposed on the beam, which makes averages away the ν-frequency components.

For nonlinear forces, the invariant curves are not ellipses. Nonlinearities caused by the external focusing forces (like higher magnetic multipoles) are different from nonlinearities caused by space charge. For external nonlinearities, the curves are elliptical near the origin and become distorted away from the origin. But nonlinearities caused by typical space-charge forces have their largest effect at the origin (assuming the beam is not hollow). We can match beams to nonlinear systems and include the effects of space charge, which is generally nonlinear. Matching to nonlinear systems requires numerical methods, which will be discussed below.

Why is matching important?

A beam matched to a periodic focusing channel has a periodic distribution function. This means that all properties of the beam are periodic. In particular, there is no emittance growth, halo formation, or particle loss. This is why it is so important to carefully match beams to accelerator structures.

A matched distribution is an equilibrium about which mismatched beams move. A match can be stable or unstable. Although instabilities are important for circular machines because the beam spends a long time in the machine, we can usually neglect instabilities in linear accelerators and their associated beamlines. But it is always important to avoid mismatches even in stable systems (which is usually what we have). Mismatches can result in a rapid emittance growth in a time equal to a quarter of the plasma period. For space-charge dominated beams, this time can be substantially less than the betatron period.

The concept of matching is also an important tool for deriving scaling laws. Scaling laws are relations between beam parameters (current, beam radius, emittance, etc.) and

machine parameters (focusing strength, period length, etc.). We get these relations by assuming the beam is matched to the machine. Usually, we use simple models that that describe only low-order behavior and are not exactly self-consistent. As an example, assume our beam bunches are uniformly-charged spheres. It is useful to define the space charge parameter μ as the ratio of the space-charge force constant to the external focusing force constant

$$\mu = -\frac{k_{sc}}{k_{ext}}. \tag{16}$$

We can get the following scaling laws

$$\mu \sim \frac{IR}{\epsilon^2} \sim \frac{I}{\epsilon^{3/2} k_{ext}^{1/4}}, \tag{17}$$

where I is the beam current, R is the matched beam radius, and ϵ is the emittance. Since a low value of μ is desirable (space-charge forces are likely to be nonlinear and lead to emittance growth because of the difficulty in matching to nonlinear systems), we see that large currents, small emittances, weak focusing, and large beams are all undesirable. In deriving the above simple formulas, we assumed the matched beam size is not affected by the current but the results are qualitatively the same even in the space-charge-dominated regime.

COURANT-SNYDER INVARIANTS

In this section, we restrict ourselves to linear motion in 1-D. If we describe an ellipse in phase space by

$$\gamma x^2 + 2\alpha x p_x + \beta p_x^2 = \epsilon, \tag{18}$$

then the ellipse changes in time, but the following quantities

$$\begin{aligned} \gamma\beta - \alpha^2 &= \text{const.} \\ \epsilon &= \text{const.} \end{aligned} \tag{19}$$

remain invariant. This makes sense once we realize that the area of an ellipse is $\pi\epsilon/\sqrt{\gamma\beta-\alpha^2}$. Eq. (19) is true for any ellipse, not just the invariant (matched) ellipse. For the invariant ellipse, Eq (18) is known as the Courant-Snyder invariant. We usually fix the normalization of the β function by setting $\beta\gamma - \alpha^2 = 1$ so that the area becomes simply $\pi\epsilon$.

For a fourth-degree curve,

$$c_0x^4 + c_1x^3p_x + c_2x^2p_x^2 + c_3xp_x^3 + c_4p_x^4 = \epsilon^2, \tag{20}$$

the analog of the first of eqs. (19) is

$$c_0c_4 - 4c_1c_3 + 3c_2^2 = \text{const.} \tag{21}$$

We can verify this fact by transforming the x and p_x in Eq. (20) by a general linear map of unit determinant and rearranging terms to determine the new coefficients. In general, there are no invariant fourth-degree curves for linear systems, expect for certain special cases. Beams matched to linear forces must have elliptical symmetry.

MOMENTS OF BEAMS MATCHED TO LINEAR SYSTEMS

The distribution matched to a periodic, linear focusing system is of the form given by eq. 15. where F is any function. The moments of such a distribution depend on the Courant-Snyder parameters and the arbitrary function F. The result for $<x^2>$ is

$$<x^2> = \frac{\int_0^\infty d\epsilon\,\epsilon F(\epsilon)}{\int_0^\infty d\epsilon\, F(\epsilon)}\ \frac{\beta}{2} \tag{22}$$

with similar expressions for the other two second moments. We can eliminate the appearance of the function F by writing these relations in terms of the rms emittance

ϵ_2.

$$\begin{aligned} <x^2> &= \beta\,\epsilon_2 \\ <xp_x> &= -\alpha\,\epsilon_2 \\ <p_x^2> &= \gamma\,\epsilon_2 \end{aligned} \tag{23}$$

These are relations among the three second moments that must be satisfied for the beam to be matched to second order. (These are the same formulas used in particle simulation codes for fitting particle distributions to ellipses.)

We can do a similar thing to get the fourth moments of the matched distribution. For $<x^4>$ we get

$$<x^4> = \frac{\int_0^\infty d\epsilon\,\epsilon^2 F(\epsilon)}{\int_0^\infty d\epsilon\,F(\epsilon)}\,\frac{3\beta^2}{8} \tag{24}$$

with similar expressions for the other four fourth moments. We can eliminate the function F from these expressions by writing the moments in terms of the fourth-order emittance ϵ_4.

$$\begin{aligned} <x^4> &= \frac{\sqrt{3}}{2}\,\beta^2\,\epsilon_4^2 \\ <x^3p_x> &= -\frac{\sqrt{3}}{2}\,\alpha\beta\,\epsilon_4^2 \\ <x^2p_x^2> &= \frac{1}{2\sqrt{3}}\,(1+3\alpha^2)\,\epsilon_4^2 \\ <xp_x^3> &= -\frac{\sqrt{3}}{2}\,\alpha\gamma\,\epsilon_4^2 \\ <p_x^4> &= \frac{\sqrt{3}}{2}\,\gamma^2\,\epsilon_4^2 \end{aligned} \tag{25}$$

These are the correct values for the fourth moments of a distribution matched to a linear system. For a distribution to be exactly matched, moments of all orders must have the right values. If the forces are nonlinear, which will almost always be the case if space charge is appreciable, then we can compute the matched values of moments numerically, using a moment simulation code. These distributions will not have elliptic symmetry.

NUMERICAL SIMULATION BY MOMENTS

It is easy to compute the evolution equations for the moments. For the moment $<xp_x>$ we have

$$\begin{aligned} \frac{d}{dt}<xp_x> &= <(\frac{d}{dt}x)p_x> + <x(\frac{d}{dt}p_x)> \\ &= \frac{1}{m}<p_x^2> + <xF_x> \\ &= \frac{1}{m}<p_x^2> + <x(a_0 + a_1x + a_2y + a_3z + \cdots)> \\ &= \frac{1}{m}<p_x^2> + a_0<x> + a_1<x^2> + a_2<xy> + a_3<xz> + \cdots. \end{aligned} \tag{26}$$

Using a polynomial approximation for the force ensures that the right-hand side is a function of moments. The evolution equations for all the moments are similarly derived. We can truncate the moment equations at some maximum order, yielding a finite system of equations. One feature of this approach is the consistency between the description of the beam and the forces. That is, if we raise the maximum order of our system of moment equations, we include higher force nonlinearities and also describe the beam in greater detail (higher moments). The truncated system of equations can be integrated numerically, so we can compute the values of the moments at any time, given their initial values. This moment-simulation idea was introduced by Paul Channell[5]. We have been working on an experimental moment simulation code at Los Alamos called BEDLAM[6], which is a fourth-order[3], 3-D code[4]. As shown in fig. 7, BEDLAM computes the final moments, given the initial moments and the external focusing forces. The external forces are specified by a table of potential-expansion coefficients at each time step. The space-charge potential expansion is computed internally by BEDLAM at each time step from the spatial moments.

[3] This is equivalent to a third-order conventional optics code because an n-th order moment code includes forces up to order $n-1$.

[4] A 2-D moment simulation code, suitable for electron or ion beams, is described in ref. [7].

The moment-evolution equations are not in the form of Hamilton's equations (the moments cannot be paired into coordinates and conjugate momenta). This means we cannot use symplectic integrators to solve the differential equations. However, the moment equations do have a Lie-Poisson structure, which is related to the Poisson bracket. We have developed Lie-Poisson integrators[10] that preserve this bracket structure exactly. This provides the numerical stability analogous to that provided by symplectic integrators for single-particle motion.

A moment code is more efficient than a particle code because of the small number of equations (one for each moment) that have to be integrated. This difference is especially important in 3-D. Suppose that a 2-D particle code has a 20×20 mesh on which the space-charge forces are computed. A 3-D code of equivalent accuracy would require a 20×20×20 mesh, which means there are twenty times as many field equations to solve and also twenty times as many particles required to fill the bins. So it takes about twenty times as much work to the the 3-D calculation compared to the 2-D calculation.

Now look at the situation for moment codes. Table 1 shows the total number of moments for orders 2 through 8 for both 2-D and 3-D. We see for a fourth-order simulation, for example, that the number of moments in 3-D is only about three times as many as in 2-D.

HIGHER-ORDER SPACE-CHARGE EFFECTS

The moment approach used in BEDLAM makes more clear (compared to the particle approach) how space-charge effects modify the beam behavior as we go from linear to nonlinear motion. Let us compare here what goes on in a second-order moment simulation (using the TRACE3D code[8]) to what happens in a fourth-order simulation.

Below, we assume 1-D for simplicity, although both the TRACE3D and BEDLAM codes are actually 3-D codes.

In both cases we start the space-charge computation by determining the parameters of a model charge distribution. In the case of TRACE3D, we assume a uniformly-charged ellipsoid and choose the parameters of the ellipsoid by making its second moments equal to those of the beam being simulated (remember that all the moment code knows about the beam is a collection of moments). In the case of BEDLAM, we use a more-complicated 31-parameter model and match spatial moments up to fourth order. At this point we have a charge distribution. Its corresponding field we call the model field. Now the problem is to get a polynomial approximation to the model field.

In TRACE3D, we want a linear approximation

$$eE_x = k_{11}x, \tag{27}$$

where the space-charge force constant k_{11} is determined by

$$k_{11}<x^2> = <xE_x>, \tag{28}$$

where E_x is the model field. The quantity $<xE_x>$ is what appears in the moment evolution equations (see Eq. (26)) so we are choosing k_{11} to make $<xE_x>$ come out right. This approach is based on the work of Sacherer[9]. We have found that these ideas can be extended to higher order.

In BEDLAM, we want a cubic field

$$eE_x = k_{31}x + k_{32}x^2 + k_{33}x^3. \tag{29}$$

We choose the constants k_{31}, k_{32}, and k_{33} by requiring

$$<x^2>k_{31} + <x^3>k_{32} + <x^4>k_{33} \;=\; e<xE_x>$$

$$<x^3>k_{31} + <x^4>k_{32} + <x^5>k_{33} = e<x^2 E_x> \qquad (30)$$

$$<x^4>k_{31} + <x^5>k_{32} + <x^6>k_{33} = e<x^3 E_x>,$$

where E_x is the field of the BEDLAM charge model. In both cases, we are really making least-squares fits, weighted by the particle distribution, of the linear or cubic approximations to the model field.

The cubic approximation is, of course, a better approximation to the actual field than is the linear approximation. This is illustrated schematically in fig. 8. The important feature here is that linear part of the cubic fit is larger than that for the linear fit, i.e., $k_{31} > k_{11}$. This difference can be a large effect and leads to qualitative differences in beam behavior between second and fourth orders.

The space-charge parameter μ is the ratio of the space-charge force constant (k_{31} or k_{11}) to that of the external focusing force constant. In the linear model, a beam close to the space-charge limit will have a μ value close to but less than unity. The same beam using the cubic model (or the actual space-charge force) can have a $\mu > 1$. Because of this behavior, it is difficult to compare linear to nonlinear motion. For example, space-charge tune depression is not defined for $\mu > 1$ (the tune-depression factor is $\sqrt{1-\mu}$). Even though μ is larger than unity, the motion can be stable, consistent with the smaller-than-unity value of μ given by the linear model, in which the whole beam is modeled by a different, smaller, effective space-charge force constant. Fig. 10 shows the phase-space trajectories for such a situation. For small $|x|$, the orbits look locally hyperbolic, a characteristic of unstable motion. However, the overall motion is stable because the orbits enclose the origin.

We can see this effect more physically by comparing a uniform beam with a more

realistic, nonuniform distribution, such as a Gaussian. Fig. 9 shows charge density ρ as a function of r for a uniform sphere and a spherical Gaussian. Both beams have the same total charge and second moments (equal to 1). The graph is to scale. The central density of the Gaussian is about three times that of the equivalent uniform beam. One consequence of this factor-of-three effect is that current-limit formulas (or other scaling laws based on linear models) may not be reliable even in the moderate-space-charge regime of $\mu \approx 0.5$.

HIGH-ORDER NUMERICAL MATCHING USING MOMENTS

We have tested the idea of computing matched beams using a moment code[6]. This was done by a matching code that runs as a separate process from the BEDLAM code. The matching code reads the final moments computed by BEDLAM and generates a new file of input moments, which can be used by BEDLAM. Our present matcher code is trivial (it simply averages the input and output moments of the previous BEDLAM run to generate the new input moments) but works well. To compute a matched beam, we just execute BEDLAM and the matching code alternately until the moments stop changing. Fig. 11 shows this process. This approach of using a separate code to do the matching is practical because because the amount of data being moved is small (203 numbers). For our example, we took a permanent-magnet quadrupole channel without space charge. To enhance nonlinear effects, we used short magnets with small bores. With the nonlinearity of the magnet fringe fields temporarily turned off, we found that we obtained the same matched beam found in the conventional approach using the TRACE3D code. We used the relations given in eqs. (23) to get the Courant-Snyder parameters from our computed moment values. Of course, we obtained more

information than in the conventional approach. We got the matched values of all the moments up to fourth order.

We found that using this matched-to-the-linear-system beam was no longer matched when we turned the nonlinear fringe fields back on. (The initial moments were not equal to the final moments.) Moveover, we found that the rms emittances were not preserved in this case. Rematching to the nonlinear forces, however, made the input and output moments equal and preserved the rms emittances. We should point out that the introduction of the nonlinearities changed all the moment values. Thus, even though we did not change the linear forces when we turned on the nonlinearities, the matched values of even the second moments (and thus the Courant-Snyder parameters) were changed.

3-D SPACE-CHARGE MODEL IMPROVEMENTS

The space-charge problem is to, at every time step in the simulation, take the vector of spatial moments

$$\pi = (<x^2>, <xy>, <xz>, \ldots <z^4>), \tag{31}$$

and from it to compute a polynomial approximation to the space-charge potential

$$\phi_{sc}(x, y, z) = u_1 x^2 + u_2 xy + \cdots + u_{31} z^4. \tag{32}$$

We are interested only in methods that are efficient, for example, we would like to go from moments to potential expansion by doing just a single matrix multiply

$$u_i = \sum_{j=1}^{31} M_{ij} \pi_j, \quad i = 1, \ldots, 31 \tag{33}$$

at each time step. If the matrix M is fixed and not computed at each time step, such a computation will be extremely fast. This approach promises to make possible not just

simulations but optimization codes which include high-order, 3-D space charge effects. This is a capability that is clearly not achievable with particle-in-cell codes without vast improvements in computing power.

Recently, Paul Channel introduced some new ideas that have been investigated by Hyo Ahn. We present here some of the preliminary results.

Instead of using a 31-parameter space-charge model, we consider a fixed collection of a large number N (several thousand) points in space (let's call it a mesh). Now consider a point charge at each mesh point. Adjust the magnitudes of all the charges so that all the moments up to fourth order are the same as those of the beam we are simulating. This is an extremely undetermined system. To get a solution, we minimize the sum of the squares of the charges at the points

$$\sum_{i=1}^{N} a_i^2, \tag{34}$$

subject to the constraint that the moments are exactly right. This method selects, from the infinite number of distributions that have the correct moments, the one that has maximum smoothness. This seems like a reasonable approach but there is significant improvement we can still make. The problem is that this procedure tends to spread charge over as much space as it can and some of the charges are negative. Instead of minimizing Eq. (34), we minimize

$$\sum_{i=1}^{N} (a_i - \mu_i)^2, \tag{35}$$

where μ_i is a constant vector of N numbers. If we set the μ vector so that it resembles a reasonable beam (i.e., concentrated near the origin), then this process generates a charge distribution that is close to our reference beam μ_i but satisfies the moments exactly. This procedure was tested and produces good fits to a wide variety of beams.

The second part of the problem, now that we have the model charge distribution, is to fit its field to a polynomial potential. One way to choose the potential coefficients u_i is to do a least-squares fit of the polynomial to the points on the mesh, i.e., we try to make the value of the polynomial approximation as close as possible to the potential of the model charge distribution over the whole region. Actually, we get much better results if we weight the fitting with the values μ_i. Ideally, we want to weight by the actual charge. That is, we want good accuracy in the polynomial approximation where most of the charge is, and it is less important to be accurate where there are not many particles. Unfortunately, it is not possible to get an algorithm of the form of eq. (33) when we do this. However, using μ_i for the weight vector works fairly well and is of the form of eq. (33). In a practical simulation code, we would precompute a collection of matrices M, each corresponding to some different beam size. Whenever the beam size in the simulation changes appreciably, we switch to a new matrix, closer to the present beam size.

There are 31 potential coefficients to choose. When we do the fitting, we can do so by making the 31 quantities $<m_i\phi>$ correct, where the m_i are the phase-space monomials of orders two through four. However, the moment evolution equations really depend on the quantities that we call the generalized Sacherer integrals $<m_iE_j>$, where here the m_i are the monomials of orders one through three and E_j are the field components in the three directions. The problem is that there are 57 such generalized Sacherer integrals and only 31 potential coefficients to vary. We cannot make all the Sacherer integrals exactly correct. What we have found is that when we do field fitting instead of potential fitting, we can get the Sacherer integrals like $<xE_x>$, $<x^2E_x>$, and $<x^3E_x>$ exactly correct,

but not terms integrals like $<xyE_x>$ in which the monomial contains coordinates in directions different from the field direction. Our procedure gets 31 quantities correct and some of them are the Sacherer integrals, as mentioned above. The other quantities that it gets correct are things like $<xE_y + yE_x>$, which are not Sacherer integrals. Fortunately, in real cases, such quantities are much smaller than the Sacherer integrals that we do get right. The distinction between potential and field fitting is probably only important for a fourth-order code. At higher orders, the difference between the potential and field fit is not great.

FUTURE DIRECTIONS

Our future work will concentrate on developing techniques for designing nonlinear matching sections. Also, we plan to study the feasibility of fundamental improvements in the moment approach that will enable us to study the beam halo problem in a new way.

Nonlinear matching sections

We have seen that because the moments are a description of the distribution, because of the efficiency of the moment simulation, and because of the small amount of data that has to moved around, it is easy to compute beams matched to periodic linear or nonlinear focusing systems. In a similar way, we believe it should be possible to design nonlinear matching sections that take a given beam and transform it into one that matches a given structure. Just as we had an external matcher code, we can use an external optimizer code that varies the external-force file (again, a relatively small data set) and adjusts the forces (beamline elements) so that the final moments come out correct.

An important use of a nonlinear matching section would be to correct a beam generated by a linear accelerator before it is injected into a storage ring. The storage ring is a periodic, very-nearly linear device but the beam we are injecting has been subjected to unavoidable space-charge nonlinearities at the low-energy end of the machine. A nonlinear matching section will be able to can adjust both the core and halo regions of the beam, undoing, to some extent, the harmful effects of previous nonlinear effects.

New dynamical variables

It may be useful to examine if the moment invariants (or some functions of them) can be used as dynamical variables instead of the moments themselves. An advantage of this approach is that, in the absense of nonlinearities, the variables will be constant in time. Thus we will be solving directly for nonlinear effects. This is analogous to using amplitude-phase (action-angle in the Hamiltonian framework) variables in single-particle motion. We know that the higher-order emittances ϵ_n are more sensitive to the beam halo than the lower-order emittances. Using the moment invariants as variables will be advantageous in studying the behavior of beam halo.

Actually, if we interested in halo motion, it may be even more useful to investigate the idea of some kind of "halo variables." The philosophy of the moment method is that we make the things we are interested in (properties of the distribution) into the dynamical variables instead of trying to extract this information from single-particle motions. So some of our variables should directly measure the amount of beam in the halo, and we should compute the evolution of this quantity directly. We know that we can use other basis functions besides the monomials to get new variables. However, it is difficult (maybe impossible) to make the truncated system of new moment equations

into a Lie-Poisson system, which is what is required to preserve the Hamiltonian nature of the motion. This is a concern because we use Lie-Poisson integrators in moment simulation codes. A reasonable approach would be to study to what extent we can live with the problem of not preserving the Hamiltonian structure in our simulations. (This is like working with nonsymplectic integrators; usually not desirable but possible to do.) For short times, this should work. We should be able to study effects in which halos are generated quickly. The advantage of doing this without using particles is that the fraction of particles in the halo is low. The halo-variable approach overcomes the statistical problem of particle codes.

REFERENCES

[1] F. Neri and G. Rangarajan, Phys. Rev. Letters 64 (1990) 1075.

[2] W. P. Lysenko and M. S. Overley, in: Linear Accelerator and Beam Optics Codes, AIP Conference Proceeding 177, ed. C. R. Eminhizer (American Institute of Physics, NY, 1988) p. 323.

[3] D. D. Holm, W. P. Lysenko, and J. C. Scovel, J. Math. Phys. 31 (1990) 1610.

[4] A. J. Dragt, F. Neri, G. Rangarajan, Phys. Rev. A45 (1992) 2572.

[5] P. J. Channell, IEEE Trans. Nucl. Sci., 30 (1983) 2607.

[6] W. P. Lysenko, in: Computational Accelerator Physics, AIP Conference Proceeding 297, ed. R. Ryne (American Institute of Physics, NY, 1994) p. 517.

[7] K. T. Tsang, C. Kostas, and A. Mondelli, in: Computational Accelerator Physics, AIP Conference Proceeding 297, ed. R. Ryne (American Institute of Physics, NY, 1994) p. 485.

[8] K. R. Crandall, TRACE 3-D Documentation, Los Alamos National Laboratory report LA-11054-MS (August 1987).

[9] F. J. Sacherer, IEEE Trans. Nucl. Sci., 18(1971) 1105.

[10] P. J. Channell and J. C. Scovel, Physica D50 (1991) 80.

order	number of moments	
	2D	3D
2	10	21
4	65	203
6	205	917
8	490	2996

Table 1. The number of moments from second to the given order for 2-D and 3-D. We see that the number of moments required to describe the beam in 3-D is only several times as many as for 2-D.

order n	ϵ_n (uniform)	ϵ_n (hollow)
2	0.250	0.500
4	0.380	0.685
6	0.462	0.734
10	0.567	0.811
20	0.697	0.886
100	0.895	0.968
400	0.964	
1000		

Table 2. The n-th order emittances for a uniformly-filled disk and a hollow distribution, both with a total emittance of 1. For high orders, ϵ_n is insensitive to the interior of the distribution and tends to measure the total area in phase space.

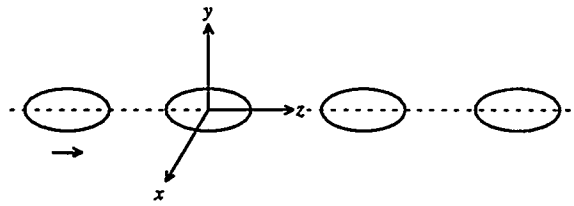

Fig. 1. Coordinates move with the beam bunch and measure displacement relative to the bunch center.

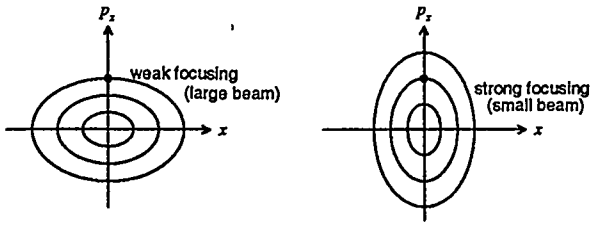

Fig. 2. For time-independent, linear focusing forces, the phase-space trajectories are the invariant ellipse, which are constant in time.

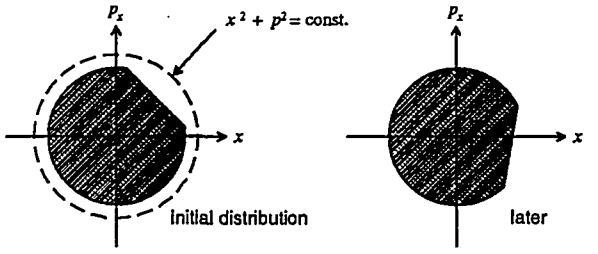

Fig. 3. For time-independent, linear forces we can scale the variables so that the invariant ellipses are circles. The evolution of any distribution is then a rotation.

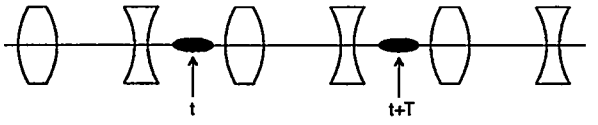

Fig. 4. A bunch traveling down an alternating-focusing lattice sees forces periodic in time.

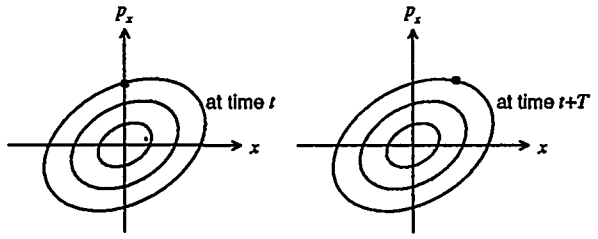

Fig. 5. For period, linear focusing, certain ellipses (the invariant ellipses) are unchanged in one period, even though individual points on the ellipses are at different locations.

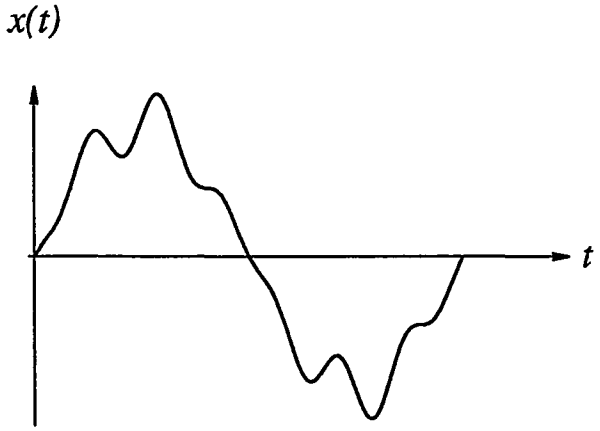

Fig. 6. The solution for a periodic focusing system in which the phase advance is 60° per period.

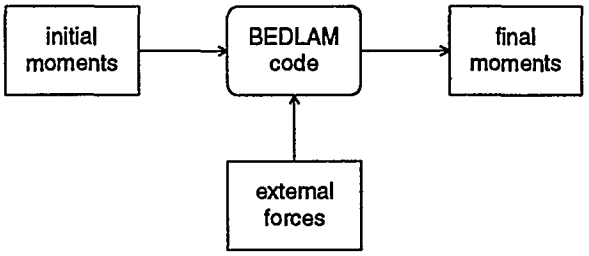

Fig. 7. The 4-th order, 3-D moment simulation code BEDLAM computes the moments at any given time, given the initial moments and a table describing the focusing forces. The space-charge forces are determined by BEDLAM in terms of the spatial moments at each time step.

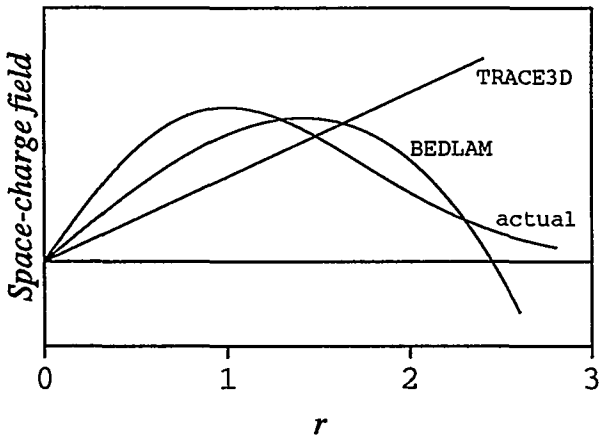

Fig. 8. For a typical charge distribution, BEDLAM's cubic-force model is more like the actual space-charge force than the linear model in TRACE3D.

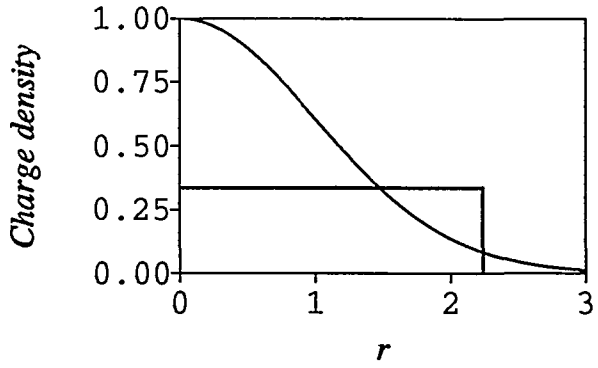

Fig. 9. Gaussian and uniform charge distribution in 3-D. Both charge distributions have the same total charge and second moments ($\sigma = 1$) but very different central densities.

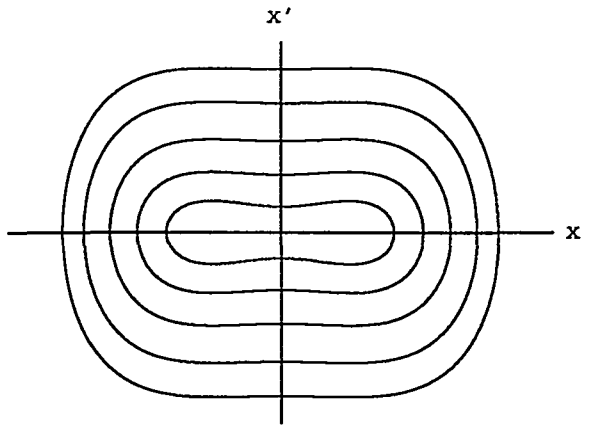

Fig. 10. Motion in phase space for nonuniform beams like Gaussians when $\mu > 1$. Motion for small $|x|$ appears unstable (locally hyperbolic) even though overall motion is stable (circles origin of phase space).

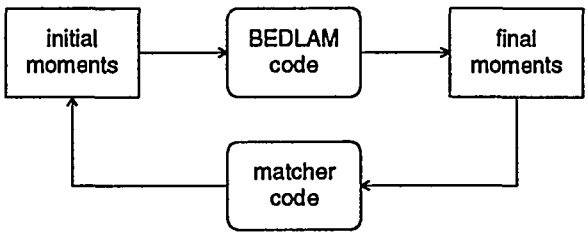

Fig. 11. Data flow for matching with BEDLAM. The BEDLAM code transports initial moments to their final values. The MATCHER code computes a suggested initial set of moments for BEDLAM to try. This process is repeated until the moments stop changing.