The $\delta f$ Algorithm for Beam Dynamics

J. Koga and T. Tajima
Institute for Fusion Studies
The University of Texas at Austin
Austin, Texas 78712

May 1993
The $\delta f$ Algorithm for Beam Dynamics

J.K. Koga$^a$ and T. Tajima
Institute for Fusion Studies
The University of Texas at Austin
Austin, Texas 78712

Abstract

An algorithm is developed to study particle dynamics of beams including collective interaction with high accuracy and low noise. Particle dynamics with collective interactions is treated through particle simulation, where the main or average distribution $f_0$ and the deviation away from it $\delta f$ are separately followed. The main distribution $f_0$ is handled by an analytic equilibrium solution and the perturbation away from it $\delta f$ is followed by the method of characteristics. We call this the $\delta f$ algorithm. We specifically model a synchrotron collider which includes the collision section where collective effects of collisions are simulated by this $\delta f$ algorithm and the rest of the collider where single particle dynamics are treated by simple harmonic transport. The most important target of this simulation is to understand and predict the long-time ($10^8 - 10^9$ rotations) behavior of the beam luminosity and lifetime. The $\delta f$ method allows the study of the effect of small perturbations over long timescales on beam lifetime by eliminating the numerical noise problem inherent in Particle-in-Cell techniques. In the $\delta f$ code using the reference parameters of the SSC (Superconducting Super Collider), beam blow-up near resonances and oscillations in the tune shift, $\Delta \nu$, far from resonances are observed. In studying long timescale particle diffusion in the phase space of the beams away from resonances, the $\delta f$ code performance is compared with a tracking code which does not incorporate collective interaction.

$^a$Superconducting Super Collider Laboratory Fellow
I Introduction

In this paper we present an algorithm which allows the study of collective particle dynamics in beams with realistic fluctuation noise levels. The so-called $\delta f$ algorithm in which the fluctuating part of the distribution function is simulated has been applied to various problems in recent years.\textsuperscript{1, 2, 3, 4} The common theme of all these studies is the reduction of the enhanced particle fluctuation noise characteristic of standard Particle-in-Cell (PIC) techniques.\textsuperscript{5, 3} The $\delta f$ algorithm allows the study of subtle plasma instabilities using a much smaller number of simulation particles than needed by the PIC technique. In this paper we apply the $\delta f$ algorithm to the beam-beam interaction which is one of the principle limitations on beam intensity in high energy synchrotron colliders.\textsuperscript{6, 7}

One of the goals of high energy particle accelerators is achieving a large number of collision events from high energy colliding beams. In circular accelerators or synchrotrons this is accomplished by colliding two focussed beams which are travelling in opposite directions. The luminosity of the colliding beams is defined as:

$$L = f \frac{N^2}{4\pi\sigma^2}$$

where $N$ is the number of particles, $\sigma$ is the rms beam size, and $f$ is the frequency of collisions. To achieve a large interaction rate, it is necessary that the luminosity $L$ be as high as possible. High luminosity is achieved by high collision frequencies, a large number of particles per beam, and small beam sizes. However, higher $N$ increases collective effects, higher $f$ results in multi-bunch instabilities, and lower $\sigma$ places more demands on focusing systems and beam sources. Typically the luminosity $L$ is a number between $10^{30}$ and $10^{33}$ cm\textsuperscript{-2} sec\textsuperscript{-1} for contemporary high energy accelerators. At high energies the interaction cross section $\sigma_{int}$ tends to be small on the order of $10^{-32}$ to $10^{-33}$ cm\textsuperscript{2}, as it is inversely proportional
to the square of the beam energy. A large number of collisions is necessary to achieve a statistically significant amount of data. For example, in the Superconducting Super Collider (SSC) the projected storage time in the main ring is 24 hours. In this amount of time the bunched beams will undergo approximately $10^8$ rotations and collisions. Therefore, the beams need to remain coherent for a long period of time making long term beam stability a major concern with circular colliders. Therefore, even small perturbations such as the beam-beam interaction can lead to beam spreading reducing beam luminosity and beam lifetimes. In the beam-beam interaction each beam imparts an impulse via their collective electromagnetic fields on the other beam at the interaction point where the beams cross. For hadron colliders the beam-beam interaction is expected to be crucial, since there is no synchrotron radiation damping to stop beam blow-up as in electron storage rings. This impulse may be treated as a kick, as the interaction time is much shorter than the beam particle dislocation time due to collisions. The kick can include both the impulse acting on whole beams and impulses acting on individual particles within each beam. Beam-beam plasma collective effects include plasma instabilities or "soft" collisions. These instabilities modify the beam profile and can contribute to increasing beam size. Collective instabilities have the most effect in the interaction region of the storage ring where the beam densities are highest in the accelerator. The relative importance of collective effects in plasmas is determined from the plasma parameter $g$:

\[ g = \frac{1}{n\lambda_D^3} \]  

(2)

where $n$ is the density and $\lambda_D$ is the Debye length. If $g \ll 1$, collective effects play an important role. For SSC type parameters $g = 2.66$. So collective effects are not dominant for a single pass beam-beam interaction. However, the effects of a large number of successive interactions have yet to be determined.

Numerical simulation of accelerator beam dynamics has a relatively short history. As
accelerators became increasingly more costly and complex, computers and computational techniques also became increasingly more developed. Computer simulation has recently become an accepted standard method of investigation of accelerators. It certainly is this way for the Tevatron. For the SSC one may say even that is has become one of the central design techniques. The simulations allow the study of the problem under very controlled conditions with accuracy limited by the precision of the computer. Analytical methods provide a means to study the problem in the linear regime. However, nonlinear aspects are not easily accessible. Numerical methods allow the study of this regime with fewer approximations than analytic methods. Simulation schemes such as the PIC methods\textsuperscript{5,3} represent a medium ground between the 2 particle picture of the beam-beam interaction and the full statistical picture representing all particles in the beams. However, the enhanced particle fluctuation noise in PIC methods presents a problem when studying a small perturbation such as the beam-beam interaction over long timescales. By employing the $\delta f$ model at the interaction point, an assessment of the relative importance of collisions as a whole and individual "soft" collisions (collective effects) can be determined without the problem of fluctuation noise.

In the following sections the numerical codes used to study the beam-beam interaction will be described. In all the codes a one dimensional model using the tracking or $\delta f$ technique is employed at the interaction point so that oscillations in only one transverse direction due to the counterstreaming beams are studied. The rest of the machine is treated by simple harmonic transport (betatron oscillations). We will examine the contribution of self consistent effects on beam blow-up and particle diffusion after a large number of interactions.

\section{Tracking code}

In this section we describe the tracking code technique.\textsuperscript{8} The tracking code results are used as a comparison with $\delta f$ simulation results.

The basic principle of tracking codes is to follow the dynamics of single particles around
the machine. In the beam-beam interaction the tracking code consists of two components: a target beam and a projectile beam. The target beam is treated as a rigid smooth Gaussian distribution of a large number of particles. It remains unchanged from interaction to interaction. The projectile beam is considered to be a collection of mutually noninteracting particles which are perturbed by the target beam. This is the so called “weak-strong” approximation. In tracking code simulations in the “weak-strong” approximation, transport about one turn is simulated as the product of two matrices, one for the one turn Courant-Snyder map, and the other for the impulsive application of the beam-beam interaction discussed above:

\[
\begin{bmatrix}
  x \\
  x'
\end{bmatrix}_{\text{final}} = M 
\begin{bmatrix}
  x \\
  x'
\end{bmatrix}_{\text{initial}},
\]

(3)

\[
M = \begin{bmatrix}
  \cos(2\pi\nu_0) & \beta_0^* \sin(2\pi\nu_0) \\
  -\sin(2\pi\nu_0) / \beta_0^* & \cos(2\pi\nu_0)
\end{bmatrix} \begin{bmatrix}
  1 \\
  4\pi \Delta\nu_0 F(x) / \beta_0^* \\
  0 \\
  1
\end{bmatrix}
\]

(4)

where \( x \) is the position of the particle, \( x' \) is \( dx/ds \), \( s \) is the distance along the collider, \( \nu_0 = \int ds / \beta(s) \) is the tune, \( \Delta\nu_0 \) is the input tune shift, \( \beta_0^* \) is the betatron oscillation amplitude at the interaction point (IP), and \( F(x) \) is the 1-D truncation of the force from a round Gaussian beam

\[
F(x) = \frac{1 - \exp(-x^2/2\sigma_{x0}^2)}{x^2/2\sigma_{x0}^2},
\]

(5)

where \( \sigma_{x0} \) is the beam standard deviation in \( x \). This formulation is similar to that of Neuffer et al.; however, here both beams are of the same charge. For comparison with one dimensional simulation results, \( F(x) \) becomes the force of a 1-D Gaussian slab:

\[
F(x) = \sqrt{\frac{\pi}{2}} \left( \frac{\sigma_{x0}}{x} \right) \text{erf} \left( \frac{x}{\sqrt{2\sigma_{x0}}} \right)
\]

(6)

where \( \text{erf} \) is the error function.

The first matrix in Eq. (4) takes into account the particle motion from the lattice magnets. The second matrix takes into account the kick from the beam-beam interaction.
III The $\delta f$ algorithm

We describe a collider model using the $\delta f$ method. In many problems involving subtle plasma instabilities developments in particle phase space are significant only for a small portion of the total number of particles and those for a large portion of particles may be more readily described. We propose a numerical algorithm to pinpoint the small contributions within the particle-code framework, by isolating the more interesting part of the distribution. In this model the collider is broken into two sections. One section models the interaction region. The other section models the rest of the storage ring. In the interaction region it is necessary to take into account the beam-beam interaction. Since self-consistent effects play an important role in the beam dynamics there, the $\delta f$ method is used. The rest of the collider is modelled using the Courant-Snyder map which simply involves a symplectic rotation of the particles in phase space.$^9$

Our model differs from previous models of the beam-beam interaction$^{10,8,11}$ where the beam-beam interaction is approximated as either a two particle interaction, a single particle-many particle interaction ("weak-strong" approximation),$^{10,8}$ or a many particle-many particle interaction ("strong-strong") where the beam is constrained to be a Gaussian.$^{11}$ Using a $\delta f$ code in the beam-beam model allows a many particle-many particle interaction with internal degrees of freedom in the beam shapes.

The steps of the simulation for one turn in the collider are:

1. interaction region
2. reset of fields to 0
3. symplectic mapping.

These steps are repeated until the necessary number of turns are attained. Figure 1 shows the basic geometry used in the simulation models.
Particle-in-cell (PIC) codes typically use macroparticles to represent the entire distribution of particles. In the beam-beam interaction for the SSC, the beams consist of $10^{10}$ particles each. Simulating this many particles with the PIC technique is computationally prohibitive. With the conventional PIC code $10^{10}$ particles are represented by only $10^{3} - 10^{4}$ simulation particles allowing simulation of the beam-beam interaction in a reasonable computation time. However, the fluctuation level of various quantities such as the beam density $\rho$ in the code is much higher than that of the real beam. The fluctuation level $\delta \rho$ goes as approximately:

$$\frac{\delta \rho}{\rho} \approx \frac{\sqrt{N}}{N},$$

where $N$ is the number of particles. Therefore, the fluctuation level of the PIC code is about $10^{3}$ times higher than that of the real beam. Although this probably is not significant for beam blow-up near resonances, the higher fluctuation level has a large effect on more-subtle phenomenon such as particle diffusion. The purpose of the $\delta f$ algorithm is to facilitate the study of subtle effects.$^{1, 2, 3, 4}$

The $\delta f$ method follows only the fluctuating part of the distribution instead of the entire distribution. This is essentially modelling the numerator on the right hand side of Eq. (7). So the $10^{3} - 10^{4}$ computational particles are used to represent $\sqrt{10^{10}}$ or $10^{5}$ real fluctuation particles in SSC beams. This is only one or two orders of magnitude beyond the number of computer particles.

PIC strong-strong codes use a finite number of particles to represent the Vlasov equation or Klimontovich equation.$^{12}$ In the particular case of the beam-beam interaction:

$$\frac{\partial f}{\partial s} + x' \frac{\partial f}{\partial x} - (K(s)x - F(x, s)) \frac{\partial f}{\partial x'} = 0,$$

where $K(s)x$ is the usual magnetic guiding force and $F(x, s)$ is the beam-beam force

$$F(x, s) = \frac{2eE_x(x)}{\gamma m v^2} \delta_p(s),$$
where \( E_x(x) \) is calculated from the distribution of the particles and \( \delta_p(s) \) the periodic \( \delta \)-function. \( \delta_p(s) = 1 \) when \( s = nL \) where \( L \) is the accelerator circumference and \( n = 0, 1, \ldots \). The distribution function \( f \) is represented by a finite number of particles by:

\[
f(x, x', s) = \sum_{i=1}^{N} w_0 \delta(x - x_i(s)) \delta(x' - x'_i(s)),
\]

(10)

where \( N \) is the number of simulation particles used and \( w_0 \) is an initial unchanging weight assigned to each particle.

In the \( \delta f \) method only the perturbative part of the distribution is followed.\(^1\).\(^2\).\(^3\).\(^4\) The total distribution function \( f(x, x', s) \) is decomposed into

\[
f(x, x', s) = f_0(x, x', s) + \delta f(x, x', s),
\]

(11)

where \( f_0(x, x', s) \) is the steady or slowly varying part of the distribution and \( \delta f(x, x', s) \) is the perturbative part. The key to this method is finding a distribution \( f_0(x, x', s) \) which is close to the total distribution \( f(x, x', s) \). The perturbative part \( \delta f(x, x', s) \) is then small, causes only small changes to the distribution, and thus represents only the fluctuation levels. If a distribution \( f_0(x, x', s) \) close to the total distribution is not found, then \( \delta f(x, x', s) \) represents more than the fluctuating part of the total distribution defeating the purpose of the method. The ideal situation is having an analytic solution for \( f_0(x, x', s) \). In this case any numerical truncation errors which result from the necessary derivatives of this function are eliminated. If an analytic solution cannot be found, then a numerical solution needs to be found which is close to the total distribution \( f(x, x', s) \) and is slowly varying. Continual numerical update of \( f_0(x, x', s) \) would also defeat the purpose of the \( \delta f \) method, since the PIC technique essentially does this also.

For beam-beam interaction an analytic solution to an equation close to the original Vlasov equation can be found. With a linearized beam-beam force the Vlasov equation can be written in the form:

\[
\frac{\partial f_0}{\partial s} + x' \frac{\partial f_0}{\partial x} - (K(s) - F_0(s))x \frac{\partial f_0}{\partial x'} = 0,
\]

(12)
where

\[ F_0(s) = F_0 \delta_p(s) . \]  \hspace{1cm} (13)

\( F_0 \) is the linear portion of the beam-beam force \( F(x) \). The solution is a Gaussian of the form:

\[ f_0(r) = N \beta^* \frac{2\pi}{\sigma^2} \exp \left( -\frac{r^2}{2\sigma^2} \right) , \]  \hspace{1cm} (14)

where \( r^2 = x^2 + \beta^* x'^2 \), \( N \) is the total number of particles in the beam, \( \beta^* \) is the betatron oscillation length, and \( \sigma \) is in the \( x \) direction. Note that if the beam-beam force were linear this solution \( f_0(r) \) would represent the distribution for all time in the interaction region as well as in the rest of the storage ring. Only the values of \( \beta^* \) and \( \sigma \) differ between the two regions. In the interaction region the \( \beta^* \) and \( \sigma \) are calculated using the dynamic \( \beta \) model which assumes a linear beam-beam force\textsuperscript{13,14}:

\[ \cos(2\pi \nu) = \cos(2\pi \nu_0) + 2\pi \Delta \nu \sin(2\pi \nu_0) , \]  \hspace{1cm} (15)

\[ \frac{\beta^*}{\beta_0^*} = \frac{\sin(2\pi \nu_0)}{\sin(2\pi \nu)} , \]  \hspace{1cm} (16)

where \( \nu_0 \) and \( \beta_0^* \) are the unperturbed quantities valid in the rest of the storage ring and \( \nu \) and \( \beta^* \) are the quantities perturbed by the linearized beam-beam force. From the perturbed \( \beta^* \) the perturbed beam width \( \sigma \) can be calculated from the formula:

\[ \frac{\beta_0^*}{\beta^*} = \frac{\sigma_0^2}{\sigma^2} , \]  \hspace{1cm} (17)

where \( \sigma_0 \) is the unperturbed beam width which is obtained from the assumption that the beam emittance is unchanged due to this linear beam-beam force. An equation for the perturbed \( \beta^* \) can be written in terms of unperturbed quantities from Eqns. (15) and (16):

\[ \left( \frac{\beta^*}{\beta_0^*} \right)^2 - 4\pi \Delta \nu_0 \cot(2\pi \nu_0) \left( \frac{\beta^*}{\beta_0^*} \right)^{3/2} - (2\pi \Delta \nu_0)^2 \left( \frac{\beta^*}{\beta_0^*} \right) - 1 = 0 , \]  \hspace{1cm} (18)
where $\Delta \nu_0$ is the unperturbed one dimensional tune shift. Eq. (18) can be expressed in terms of the perturbed $\sigma$ using Eq. (17):

$$
\left( \frac{\sigma}{\sigma_0} \right)^4 - 4\pi \Delta \nu_0 \cot(2\pi \nu_0) \left( \frac{\sigma}{\sigma_0} \right)^3 - (2\pi \Delta \nu_0)^2 \left( \frac{\sigma}{\sigma_0} \right)^2 - 1 = 0 .
$$

(19)

Both equations can be solved for the perturbed $\sigma$ or $\beta^*$ using a root finder. Once this is obtained the other perturbed quantities, $\nu$ and $\Delta \nu$, are obtained from Eqs. (15) and (16).

Subtracting the linearized equation in Eq. (12) from the total Vlasov equation in Eq. (8), we obtain the perturbative part of Eq. (8) for $\delta f$:

$$
\frac{\partial \delta f}{\partial s} + x' \frac{\partial \delta f}{\partial x} - (K(s)x - F_0(x, s)) \frac{\partial \delta f}{\partial x'} = -(F(x, s) - F_0(s)x) \frac{\partial f_0}{\partial x'} .
$$

(20)

$F_0(x, s)$ is the kick from a Gaussian beam and $F(x, s)$ is the kick from a Gaussian beam $F_0(x, s)$ plus the perturbation $\delta F(x, s)$. As a result of using the dynamic beta model for the stationary solution $f_0$, only the nonlinear part of the beam-beam force on the right hand side of Eq. (20) is used to advance $\delta f$. The reason for choosing the particular form of the steady state solution is apparent. The right hand side of Eq. (20) is small. The terms $\frac{\partial f_0}{\partial x'}$ and $F_0(x, s)$ are calculated using the perturbed dynamic beta quantities $\beta^*$ and $\sigma$. Note that the unperturbed Gaussian field $F_0(x, s)$ is used on the lefthand side of Eq. (20) which makes the equation linear in $\delta f$. The neglected term is

$$
\delta F(x, s) \frac{\partial \delta f}{\partial x'} .
$$

(21)

which is small in our problem. A possible incorporation of this term in the algorithm is described in Sec. V.

A. Finite Particle Representation

The perturbative part of the distribution $\delta f$ (Eq. (20)) can be represented by $N$ particles (characteristics):

$$
\delta f(x, x', s) = \sum_{i=1}^{N} w_i[s, x_i(s), x_i'(s)] \delta(x - x_i(s)) \delta(x' - x_i'(s)) .
$$

(22)
This representation is similar to the PIC representation of the entire distribution function in Eq. (10). However, the weights \( w_i \) are evolving. Note that in the worst case where the \( \delta f \) weights are used to represent the entire distribution function the \( \delta f \) method simply reverts to the PIC method. Substituting Eq. (22) into the equation for \( \delta f \) advance, we obtain:

\[
\frac{d w_i}{d s} = -\frac{1}{n} \left[ (F(x, s) - F_0(s)x) \frac{\partial f_0}{\partial x'} \right]_i
\]  

(23)

where

\[
n = \left( \frac{N_c}{\Delta x \Delta x'} \right) \quad .
\]  

(24)

This density \( n \) is calculated on the assumption that the particles are distributed uniformly in phase space. The density \( n \) is constant through the entire run, since the particles follow Hamiltonian trajectories.\(^2\) This is no longer valid when radiation or dissipative forces are included.

To ensure the conservation of total charge in the system the weights are adjusted after each \( w_i \) advance so that:

\[
\sum_{i=1}^{N} w_i = 0 \quad .
\]  

(25)

In the \( \delta f \) algorithm \( x_i, x'_i \), and \( w_i \) are advanced. The advance of the extra term \( w_i \) increases the number of operations over the PIC method and leads to other numerical constraints which will be described in the next section.

The particles are distributed uniformly in \( x \) and \( p_x \) phase space in a cylindrical coordinate system \( r \) and \( \theta \) upon initialization. The coordinates \( r \) and \( \theta \) are defined in terms of \( x \) and \( p_x \) as:

\[
r^2 = \frac{x^2}{\beta_0^*} + \frac{p_x^2}{p} \quad ,
\]  

(26)

\[
\tan(\theta) = \frac{x}{\beta_0^* \frac{p}{p_x}} \quad ,
\]  

(27)

where \( \beta_0^* \) is the betatron oscillation length at the interaction point and \( p \) the particle momentum along the collider in \( s \). The maximum \( r \) value is input into the code and is broken
up into segments of length $\Delta r$. The number of particles at each $r$ value is determined by a cumulative integration method. The particular functional form is:

$$
\Delta N = \frac{N}{N_r^2} (2r - 1),
$$

(28)

where $\Delta N$ is the number of particles to be added, $N$ is the number of particles, and $N_r$ is the number of $\Delta r$ segments to the edge of the distribution. Once the number of particles between $r$ and $r + \Delta r$ is known they are distributed uniformly in $\theta$ with a random offset $\theta_{\text{ran}}$ at $r + \Delta r/2$. The initial distribution for 1000 particles is shown in Fig. V. The purpose of this method is have each particle cover an equal area of phase space.

**B. Symplectic Integration**

Results from previous runs indicate that a higher order integration scheme for the characteristic advance is necessary for the $\delta f$ algorithm. In runs where just the leapfrog scheme is used, the algorithm is inaccurate in the particle push. This higher order integration scheme for the particles is needed in the $\delta f$ algorithm because small changes to the initial distribution are being studied. In the PIC codes the numerical noise caused by the finite number of particles is larger than that produced by the numerical diffusion of the particles caused by the leapfrog integration scheme. We describe a symplectic finite difference scheme to counter the effects of numerical diffusion on the particle motion. The normal symplectic mapping is used to advance the particles with an additional perturbation term.

Without the beam-beam force term the symplectic transformation map for the characteristics with the magnetic field is just the Courant-Snyder map. Also in the case of a linearized beam-beam force the symplectic transformation map can be written with slight modifications. The map can be written in the form:

$$
\begin{pmatrix}
    x' \\
    x''
\end{pmatrix}_f =
\begin{pmatrix}
    \cos(\theta) & \beta_0^* \sin(\theta) \\
    -\frac{1}{\beta_0^*} \sin(\theta) & \cos(\theta)
\end{pmatrix}
\begin{pmatrix}
    x \\
    x''
\end{pmatrix}_i,
$$

(29)
where \( x = dx/ds \), \( s \) is the coordinate along the collider, \( \theta = \int_{s}^{f} ds/\beta_0^* \), and the indices \( i \) and \( f \) refer to the initial and final positions, respectively. This map is used at all places in the storage ring including the interaction region. Upon adding the symplectic map the particle motion is accurate to many decimal places.

A simple implementation of the beam-beam force which preserves symplecticity involves approximating the force with an impulse. Using Hill’s equation:

\[
x'' + K(s)x = \frac{F(x)}{\gamma mv^2} \delta_p(s),
\]

where the term on the right hand side of the equation is due to the beam-beam force. The mapping is the same as a tracking code with the beam-beam force:

\[
\begin{pmatrix}
  x \\
  x'
\end{pmatrix}_f = \begin{pmatrix}
  1 & 0 \\
  G(x) & 1
\end{pmatrix}
\begin{pmatrix}
  x \\
  x'
\end{pmatrix}_i,
\]

where

\[
G(x) = \frac{F_0(x)}{\gamma mv^2} \frac{1}{x},
\]

and \( F_0(x) \) is the unperturbed force due to a Gaussian beam.

In the particle advance scheme the particles are advanced first using the transfer matrix for a distance in \( \theta = \Delta s/4\beta_0^* \) where \( \Delta s = c\Delta t \). The particles are then kicked by the beam-beam field for \( \Delta s/2\beta_0^* \) and then advanced again \( \Delta s/4\beta_0^* \). The total matrix is:

\[
\begin{pmatrix}
  x \\
  x'
\end{pmatrix}_f = M(\theta)
\begin{pmatrix}
  1 & 0 \\
  G(x) & 1
\end{pmatrix} M(\theta)
\begin{pmatrix}
  x \\
  x'
\end{pmatrix}_i,
\]

\[
M(\theta) = \begin{pmatrix}
  \cos(\theta) & \beta_0^* \sin(\theta) \\
  -\frac{1}{\beta_0^*} \sin(\theta) & \cos(\theta)
\end{pmatrix},
\]

where \( \theta = \Delta s/(4\beta_0^*) \) and \( x \) used in \( G(x) \) is the intermediate \( x \) value obtained from the first transfer matrix application.

\[
G(x) = \frac{F_0(x)}{\gamma mv^2} \frac{\Delta s}{x^2}.
\]

13
Start with $x^n, x'^n, \delta f^n, \delta f^{n-1}$

2. $\delta f_{\text{predict}}^{n+1}$ from $x^n, x'^n, \delta f^{n-1}, F^n(x^n, \delta f^n)$

3. $\delta f_{\text{predict}}^{n+\frac{1}{2}} = \frac{1}{2} \left( \delta f_{\text{predict}}^{n+1} + \delta f^n \right)$

4. $x^{n+\frac{1}{2}}, x'^{n+\frac{1}{2}}$

5. $\delta f_{\text{correct}}^{n+1}$ from $x^{n+\frac{1}{2}}, x'^{n+\frac{1}{2}}, \delta f^n, F^{n+\frac{1}{2}}(x^{n+\frac{1}{2}}, \delta f_{\text{predict}}^{n+\frac{1}{2}})$

6. $x^{n+1}, x'^{n+1}$

7. Repeat steps 1-6 until the end of the interaction region

8. Rotate $x^{n+1}, x'^{n+1}$

9. Repeat steps 1-8 until the end of the simulation run

**Table I:** Steps for advance of $\delta f$ algorithm

**C. Time Advance**

The time advancement, the predictor-corrector advance scheme, is shown in Table I. The $n$ in Table I refers to the time step number. In step 2 $\delta f_{\text{predict}}^{n+1}$ is calculated from the discretization of Eq. (23):

$$w_{i}^{n+1}_{\text{predict}} = w_{i}^{n-1} - \frac{1}{n} \left[ (F^n(x_i^n, \delta f^n) - F_0(x_i^n)) \frac{\partial f_0(x_i^n, x_i'^n)}{\partial x'} \right] 2\Delta s ,$$

where $\Delta s = c\Delta t$, and $F^n(x_i^n, \delta f^n)$ is the force calculated from the unperturbed Gaussian beam $F_0(x_i^n)$ plus the perturbation force $\delta F^n(x_i^n, \delta f^n)$. $\delta f_{\text{predict}}^{n+1}$ is then calculated using Eq. (22):

$$\delta f(x, x', s)^{n+1}_{\text{predict}} = \sum_{i=1}^{N} w_{i}^{n+1}_{\text{predict}} \delta'(x - x_i(s))\delta'(x' - x_i'(s)) .$$

14
The same procedure is used in step 5 to calculate $\delta f_{\text{correct}}^{n+1}$:

$$w_{\text{correct}}^{n+1} = w_i^n + \Delta w,$$

$$\Delta w = -\frac{1}{n} \left[ \left( F_{n+\frac{1}{2}}^{n+\frac{1}{2}}(x_i^{n+\frac{1}{2}}, \delta f^{n+\frac{1}{2}}) - F_0(s) x_i^{n+\frac{1}{2}} \right) \frac{\partial f_0(x_i^{n+\frac{1}{2}}, x_i^{n+\frac{1}{2}})}{\partial x'} \right] \Delta s.$$  

(38)  

(39)

In steps 4 and 6 $x$ and $x'$ are advanced using Eq. (33). In step 8 $x$ and $x'$ are advanced using Eq. (29):

$$\begin{pmatrix} x^{n+1} \\ x'^{n+1} \end{pmatrix} = \begin{pmatrix} \cos(2\pi \nu) & \beta_0^* \sin(2\pi \nu) \\ -\frac{1}{\beta_0^*} \sin(2\pi \nu) & \cos(2\pi \nu) \end{pmatrix} \begin{pmatrix} x^n \\ x'^n \end{pmatrix},$$  

(40)

where

$$\nu = \nu_0 - \frac{\Delta s}{\beta_0^*},$$  

(41)

which takes into account the finite length of the interaction region $\Delta s$ in the phase space rotation.

**D. Force Calculation**

One spatial dimension $x$ and three velocity coordinates ($v_x, v_y, v_z$) are followed. In the force calculation two approximations are made: (1) light waves are ignored and (2) self fields (space charge effects) among particles of the same beam are negligible. Ignoring the effects of light waves can be justified for the SSC by considering the collisionless skin depth, $\lambda_c$, of the beam where:

$$\lambda_c = \frac{c}{\omega_b}$$  

(42)

$$\omega_b = \sqrt{\frac{4\pi e^2 n_b}{\gamma m_b}}.$$  

(43)

Using parameters for the SSC, $\lambda_c \gg w$ where $w$ is the width of the beam. $\lambda_c$ is the scale length over which a plasma responds to light waves. Since $\lambda_c$ is much larger than the size of the beam, particles do not strongly interact with light waves. Self-fields of the beam are
neglected, since the forces from the other beam are much larger. The ratio of the self fields to the kick fields from the other beam goes as:

\[ \text{(self fields)} \approx \frac{1}{\gamma^2} \text{(kick fields)} , \]  

(44)

where \( \gamma = 2.13 \times 10^4 \) for SSC parameters.

With the elimination of light waves the time step of the simulations can be on the order of the plasma frequency \( \omega_p \), which occurs on a much longer timescale than light waves. The time of interaction between the two beams is \( \tau_{int} = \Delta s/2c \). \( \tau_{int} \) is the time the simulation is run before the particles are rotated in phase space. With simulation time steps in units of fractions of \( \omega_p \) the time period can now be represented by \( 1 - 4 \) simulation time steps.

Each particle in the simulation has a particle shape factor \( S(x) \). \( S(x) \) is chosen to give the particles finite size, so that short wavelength oscillations are filtered out in the fields.\(^5\)\(^3\)

This reduces noise and short range collision forces. The particular form chosen is:

\[ S(x) = \frac{1}{\sqrt{2\pi} a} \exp \left( -\frac{x^2}{2a^2} \right) , \]  

(45)

where \( a \) is the finite particle size.

The macroparticles are advanced by the Lorentz force equation:

\[ \frac{d\vec{\mathbf{p}}_i}{dt} = e \int_{-\infty}^{\infty} dx S(x - x_i)(\vec{E}(\vec{x}) + \vec{v}_i \times \vec{B}(\vec{x})/c) , \]  

(46)

where \( x_i \) is the position, \( \vec{p}_i \) is the momentum, \( \vec{v}_i \) is the velocity of particle \( i \), \( S(x) \) is the particle shape factor, and \( \vec{E}(\vec{x}_i) \) and \( \vec{B}(\vec{x}_i) \) are the electric and magnetic field of the other beam, respectively. The integral over \( x \) takes into account the finite size of the particle.

The calculation of the fields can be simplified by performing the appropriate Lorentz transforms and taking into account the highly relativistic nature of the beams being studied. For a general Lorentz transformation to a frame moving at velocity \( \vec{u} \) the transformation of the fields can be written\(^15\):

\[ \vec{E} = \gamma(\vec{E}' + \vec{\beta} \times \vec{E}') - \frac{\gamma^2}{\gamma + 1} \vec{\beta}(\vec{\beta} \cdot \vec{E}') , \]  

(47)
\[ \vec{B} = \gamma (\vec{B}' - \vec{\beta} \times \vec{E}') - \frac{\gamma^2}{\gamma + 1} \vec{\beta} (\vec{\beta} \cdot \vec{B}') , \]  

(48)

where \( \vec{\beta} = \vec{v}/c \) and \( \gamma \) is the relativistic factor. Eqs. (47) and (48) can represent transformations of the fields from the frame moving with the beam \((\vec{E}', \vec{B}')\) to the lab frame \((\vec{E}, \vec{B})\). In the beam frame the beam particles only have thermal velocities. These velocities are small and randomly oriented. Therefore, only small remnant currents are present and the approximation \(|\vec{B}'| \approx 0\) can be made. Eqs. (47) and (48) become:

\[ \vec{E} = \gamma \vec{E}' - \frac{\gamma^2}{\gamma + 1} \vec{\beta} (\vec{\beta} \cdot \vec{E}') , \]  

(49)

\[ \vec{B} = -\gamma (\vec{\beta} \times \vec{E}') . \]  

(50)

Assuming the motion of the beams is in the \( z \) direction the fields can be written:

\[ E_x = \gamma E'_x , \]  

(51)

\[ B_y = \gamma \beta E'_x . \]  

(52)

Since the beams are highly relativistic \( \gamma \gg 1 \), the approximation \(|\vec{\beta}| \approx 1\) can be made. Thus, \( E_x \approx B_y \). Using this in Eq. (46), we obtain:

\[ \frac{d\vec{p}_i}{dt} \approx e \int_{-\infty}^{\infty} dx S(x-x_i) E_x(x)(1 + v_i/c) \]  

(53)

where \( v_i \) is the velocity of the beam kicked by the other beam. Again the approximation \( v_i \approx c \) can be used:

\[ \frac{d\vec{p}_i}{dt} \approx 2e \int_{-\infty}^{\infty} dx S(x-x_i) E_x(x) . \]  

(54)

Therefore, including the effects of the magnetic field kick to the beam simply involves doubling the contribution of the electrostatic field of the other beam.

The \( \delta f \) method the steady state Gaussian part of the beam distribution produces an electric field \( E_x \) of the form:

\[ E_x(x) = 4\pi e n_b \sigma_{x0} \text{erf} \left( \frac{x}{\sqrt{2}\sigma_{x0}} \right) \]  

(55)
where $n_b$ is the beam density, $\sigma_{x_0}$ is the unperturbed beam sigma, and erf is the error function. The perturbative part of the electric field $E_x$ is calculated from:

$$\frac{\partial E_x}{\partial x} = 4\pi e \int S(x - x') \rho(x') dx', \quad (56)$$

where $\rho(x)$ is the perturbative charge density and $S(x)$ is the particle shape factor. The charge density $\rho(x)$ is the accumulation of the finite size macroparticles:

$$\rho(x) = \sum_{j=1}^{N} q_j S(x - x_j), \quad (57)$$

where $N$ is the number of particles and $q_j$ is the charge of particle $j$ which is $w_j e$ where $w_j$ is the weight of the particle calculated from the $\delta f$ method. Since the charge is accumulated on a grid, Fast Fourier Transforms (FFT) can be used to transform the grid to $k$ space where manipulation is easier:

$$\rho(x) = \sum_{j=1}^{N} q_j e^{-(x-x_j)^2/2a^2}, \quad \rho(k) = q e^{-k^2a^2/2} \sum_{g} e^{-ikx_g} \sum_{j \in g} e^{-ik\delta_j}, \quad (58)$$

where a Gaussian shape factor is used $S(x - x_j) = \exp[-(x - x_j)^2/2a^2]$, the sum on $g$ is over the grid points, $a$ is the particle size, and $\delta_j$ is the distance of the particle from the nearest grid point $x_j - x_g$. The summation term with $j \in g$ is a sum over all particles $j$ in grid cell $g$.

In order to increase the accuracy, the accumulation using cubic spline interpolation$^{16,3}$ is implemented. This assignment technique allows a smoother grid assignment than lower order methods such as the subtracted dipole scheme (SUDS) or area weighting scheme.$^{5,3}$ The charge density then takes the form$^3$:

$$\rho(k) = q e^{-k^2a^2/2} \left[ \sum_{g} e^{-ikx_g} \left( \sum_{j \in g} s_1 + \sum_{j \in g-1} s_2 \right) \right. \left. - i k \sum_{g} e^{-ikx_g} \left( \sum_{j \in g} s_3 + \sum_{j \in g-1} s_4 \right) \right], \quad (59)$$
where the summation terms with \( j \in g - 1 \) are sums over all particles \( j \) in grid cell \( g - 1 \) and the \( s \) terms are the weighting factors:

\[
\begin{align*}
    s_1 &= (1 - \delta_j)^2(1 + 2\delta_j) \\
    s_2 &= \delta_j^2(3 - 2\delta_j) \\
    s_3 &= \delta_j(1 - \delta_j)^2 \Delta \\
    s_4 &= -(1 - \delta_j)\delta_j^2 \Delta .
\end{align*}
\]

The electric field in Eq. (56) can be transformed to \( k \) space using the FFT:

\[
E_x = \frac{i}{k} 4\pi e \rho(k) ,
\]

where \( \rho(k) \) is from Eq. (59). The force on the particles \( F(x) \) can be calculated from the inverse transform of Eq. (64):

The previous field calculation solves the field for periodic boundary conditions. Note that the field equation does not take into account finite charge in the system. Finite charge is included in the \( k = 0 \) term. However, this term cannot be incorporated, since one gets a division by zero. To account for finite charge in the system, the \( k = 0 \) term in \( E_x \) can be explicitly calculated:

\[
E_x^{k=0}(x) = -4\pi \rho(0) \left( \frac{L_x}{2} - x \right) ,
\]

where \( L_x \) is the length of the system and \( \rho(0) \) is the \( k = 0 \) component of the charge density which calculates the total charge in the system. By adding this field to the field calculated from Eq. (64) after the inverse transform one gets the field with vacuum boundary conditions.

**IV Simulation Results**

In this section we describe results of the study of test problems in the beam-beam interaction. Short timescale beam-beam collective effects near resonances as well as long timescale
$l \times w \times h = 7.5\text{cm} \times 10^{-3}\text{cm} \times 10^{-3}\text{cm}$

$N_B = 7.3 \times 10^9$

$T = 20 \text{ TeV protons}$

$\beta^* = 50 \text{ cm}$

$\Delta \nu_{HO} = 0.84 \times 10^{-3}$

$\nu_{HO} = 0.285$

Luminosity = $10^{33} \text{ cm}^{-2} \text{ s}^{-1}$

Lifetime = 24 hours or $10^8$ turns

Table II: SSC Parameters

collective effects far from resonances are examined using the $\delta f$ code, which are compared against the single particle tracking code results.

A. Reference Parameters

Our research is generic enough to cover the beam-beam interaction of various colliders or storage rings. We make specific reference to the parameters of the SSC. Table A. shows parameters for the SSC. Using the numbers from the table we have: $\gamma = 2.13 \times 10^4$ and $\omega_b \tau_{int} = 0.035$ where $\omega_b = \sqrt{4\pi e^2 n_b/\gamma m_b}$, $n_b = N_B/(lwh)$ is the beam density, $m_b$ is the mass of the beam particles (protons), $\gamma$ is the relativistic factor, and $\tau_{int} = L/2c$ is the interaction time of the colliding beams. The horizontal tune shift $\Delta \nu_{HO}$ is calculated for a two dimensional Gaussian beam. Since the present simulations deal with only one dimension, this quantity is recalculated. Using the equation for the one dimensional tune shift:

$$\Delta \nu_0 = \sqrt{\frac{2}{\pi}} \frac{\beta^* r_p N_B}{\gamma hw},$$

and using values from table A., the one dimensional tune shift is $\Delta \nu_0 = 2.1 \times 10^{-3}$.

A series of simulation runs is performed using the parameters described in table A.
B. Near Resonances

A good test of the $\delta f$ code is the examination of beam blow-up near strong resonances. Although the $\delta f$ code is not good for deviations far from the equilibrium which is a characteristic of beam blow-up, the onset of resonances should be observed by the code. We examine two resonances with values of the tune just above $\nu_0 = \frac{1}{2}$ and $\nu_0 = \frac{1}{4}$.

The initial distribution of particles is shown in Fig. V with $10^3$ particles in each beam. The simulation box size is $256\Delta$ where $\Delta$ is the cell size. The beam width $\omega$ is $30\Delta$ and the particle size $a$ is $\Delta$. By normalizing the code to a plasma with density lower than the beam, where $\omega_0$ is the normalization plasma frequency and $\omega_b$ is the beam plasma frequency, only $4$ simulation time steps are needed to cover the interaction region. Thus, $\omega_0 \Delta t = 0.25$ where $\Delta t$ is the simulation time step size.

Figure 3 shows the distribution of $10^3$ particles in $(x/\sigma_x, p_x/\sigma_p)$ phase space for one beam after $50$ rotations with the tune just above $\nu = \frac{1}{2}$ at $\nu = 0.5021$ and tune shift $\Delta \nu_0 = 2.1 \times 10^{-3}$. After $50$ rotations the particles are no longer uniformly distributed in $(x/\sigma_x, p_x/\sigma_p)$ space. The distribution of particles has the football shape characteristic of resonances near $\nu_0 = 1/2$. Particles near the tails of the distribution have gained a large amount of momentum. A profile in $x$ of a Gaussian distribution of particles in Fig. 4 is shown. Figure 5 shows the deviations from the Gaussian profile after 50 rotations. The center of the beam is at $x = 128\Delta$. The maximum perturbations are about $\pm 3\sigma_x$ away from the core of the beam with $|\delta f/f_0| \approx 30\%$ there. Notice that the perturbed distribution makes sense physically. There is a depletion of particles from the center of the beam and an increase in particles at about $\pm 3\sigma_x$. A quantity often calculated in accelerator physics is the normalized emittance $\epsilon_n$:

$$\epsilon_n = (\beta \gamma) \pi \frac{1}{2} \frac{1}{\beta_0^*} \sum_{i=1}^{N} \left( x_i^2 + \beta_0^* x_i^' \right)^2$$

(67)

where $\beta$ and $\gamma$ are the usual relativistic quantities, $\beta_0^*$ is the betatron oscillation length at the
interaction point, $x' = p_x/p$, $p_x$ is the transverse momentum, $p$ is the momentum along the collider path, and $N$ is the number of simulation particles. The emittance gives a measure of phase space area occupied by the beam particles. In the $\delta f$ algorithm an initial unperturbed emittance is calculated:

$$\epsilon_{n_0} = (\beta \gamma) \frac{1}{2} \frac{1}{\beta_0^*} \sum_{i=1}^{N} \left( \frac{x_i^2 + \beta_0^* x_i^2}{\beta_0^*} \right) w_{0i}, \quad (68)$$

where $w_{0i}$ is the initial unperturbed distribution function $f_0$ for particle $i$. The perturbation:

$$\delta \epsilon_n = (\beta \gamma) \frac{1}{2} \frac{1}{\beta_0^*} \sum_{i=1}^{N} \left( \frac{x_i^2 + \beta_0^* x_i^2}{\beta_0^*} \right) w_i \quad (69)$$

where $w_i$ is the time evolving perturbation $\delta f$ for particle $i$. This perturbation emittance is calculated and added to the initial $\epsilon_n$ to get the total $\epsilon_n$. Figure 6 shows the growth of the emittance for both beams as a function of rotations. It can be seen that the beams are blowing up in phase space area very quickly.

Figure 7 shows the distribution of $10^3$ particles in $(x/\sigma_x, p_x/\sigma_p)$ phase space for one beam with the tune just above $\nu = \frac{1}{4}$ at $\nu = 0.2521$ and tune shift $\Delta\nu_0 = 2.1 \times 10^{-3}$ after 400 rotations. The distribution of particles in phase space shows the characteristic square-like shape indicative of a beam near resonance of $\nu_0 = 1/4$. A profile in $x$ of a Gaussian distribution of particles in Fig. 8 shows the deviations from the Gaussian profile (Fig. 4) after 400 rotations. The center of the beam is at $x = 128\Delta$. The maximum perturbations are at the core of the beam with $|\delta f/f_0| \approx 10\%$ there. The effect of the resonance is a widening of the beam in $x$. There is a depletion of particles from the center of the beam and an increase in particles at about $\Delta x \approx \pm\sigma_x$. Figure 9 shows the growth of the emittance for both beams as a function of rotations. The beams blow up in phase space area very quickly and then saturate at about 140 rotations. The rate at which the emittance for $\nu_0 = 0.2521$ is growing is much slower than the emittance for $\nu_0 = 0.5021$. This is expected, since the resonance near $\nu_0 = 0.5$ is stronger than that for $\nu_0 = 0.25$. 

22
C. Long Time Behavior

A series of $\delta f$ simulations have been performed to determine long time characteristics of the beam-beam interaction far from resonances. The fluctuation level $\delta$ expected for the actual SSC beam is $\delta \approx 10^{-5}$ for $10^{10}$ particles. Figure 10 shows the variation with particle number of the minimum and maximum perturbations $\delta f/f_0$ for runs with 10240 rotations. We see that the maximum perturbation is nearly independent of particle number. The minimum fluctuation value decreases exponentially with increasing particle number. It can be seen that the minimum perturbation drops below $10^{-5}$ for simulations with $10^3$ particles and larger. Because $10^3$ particles could be used, rotations of $10^5$ could be run. The initial distribution of particles is shown in Fig. V. The parameters are the same as in the previous section with the exception that the tune $\nu_0 = 0.285$ and the simulation box size is $128\Delta$.

Figure 11 shows the distribution of $10^5$ particles in $(x/\sigma_x, p_x/\sigma_p)$ phase space for one beam after $10^5$ rotations. After $10^5$ rotations the particles are no longer uniformly distributed in $(x/\sigma_x, p_x/\sigma_p)$ space. Some clumping of particles has occurred and small regions contain no particles. Figure 12 shows the perturbations from the $\delta f$ code to the Gaussian profile after $10^5$ rotations. The maximum perturbations are only 0.1% of the maximum in the Gaussian profile (Fig. 4). Thus, the $\delta f$ code is still a valid approximation.

1 Tune Shift

The tune shift $\Delta \nu$ measures the strength of the beam-beam kick. As the beams expand and contract, the kick weakens and strengthens, respectively. One method for measuring $\Delta \nu$ involves a least-squares-fit to the kicks of small and large amplitude particles. We use Sands$^{18}$ expression for linear tune shift, which is valid for small amplitude particles:

$$\Delta \nu = \frac{\beta_0^*}{4\pi} \Delta K \Delta s,$$

(70)
\( \Delta K \Delta s = \frac{\Delta x'}{x}, \)  

(71)

where \( \Delta x' = \Delta p_z/p \) and \( \Delta p_z = 2eE_w(x)\Delta t. \)

Results from the least-squares-fit method for one beam are shown in Fig. 13. The fit is done for small amplitude particles \( x < 0.1 \sigma_x \) at the top of the figure and for the entire beam for the bottom of the figure. The tune shift \( \Delta \nu \) oscillates around the unperturbed values of \( \Delta \nu_0 = 2.1 \times 10^{-3} \) for small amplitude particles and \( \Delta \nu_0 \approx 1.52 \times 10^{-3} \) for all the particles. The discrepancy is due to the drop-off of \( \Delta \nu \) at large values of \( x \). The amplitude of the variation in \( \Delta \nu \) for small amplitude particles is approximately \( \pm 3\% \) of \( \Delta \nu_0 \) throughout the run. The maximum variation of \( \Delta \nu \) is approximately \( \pm 4\% \). The oscillations in \( \Delta \nu \) indicate expansion and contraction of the beams. Notice that the small amplitude \( \Delta \nu \) of the beam has constant oscillation amplitude whereas \( \Delta \nu \) for the entire beam is increasing in oscillation amplitude for the first 15000 rotations.

D. Particle Diffusion

In this section we examine particle diffusion brought about by the beam-beam interaction. We compare the diffusion coefficients measured from the tracking code and the \( \delta f \) code.

Diffusion is determined using the method of Chirikov.\(^6\) Diffusion coefficients, \( D_1 \) and \( D_3 \), for sample particles initially distributed randomly in phase space are calculated over two different timescales.\(^6,7\) If \( D_1 \approx D_2 \), then the motion \( x \) is diffusive. If \( D_1 \gg D_2 \) where \( D_1 \) is calculated on shorter timescales than \( D_2 \), then the motion is not diffusive.

1 Tracking Code Results

We first examine particle diffusion for particles tracked using the one dimensional tracking code described in Sec. II.

Reference parameters described in Sec. A. for the SSC are used: The tune \( \nu_0 = 0.285 \) and
the tune shift $\Delta \nu_0 = 2.1 \times 10^{-3}$. Diffusion coefficients, df1 and df2, calculated after 10240 rotations are shown in Fig. 14 where $r/\sigma$ is the distance in phase space from the center of the beam

$$\frac{r}{\sigma} = \sqrt{\frac{x^2}{\sigma_x} + \frac{p_x^2}{\sigma_p}}. \quad (72)$$

The $D_x$ means that df1 and df2 are calculated for diffusion in position $|x|$. The diffusion is normalized to $\sigma_x^2/N_r$ where $N_r$ is the number of rotations. In Fig. 14 it is apparent that df1 $\ll$ df2 for all particles indicating that the motion is largely oscillatory in phase space. The coefficients calculated over two time scales differ on average by about a factor of 100. This is expected for oscillatory motion:

$$\frac{\text{df2}}{\text{df1}} \propto \left(\frac{\Delta N_1}{\Delta N_2}\right)^3$$

$$\propto \frac{1}{1000} \quad (73)$$

where $\Delta N_1 = 102$ and $\Delta N_2 = 1024$. Figure 15 shows the diffusion coefficients calculated for 40960 rotations. The range of coefficients for 40960 rotations is between $10^{-9}$ and $10^{-14}$. There are some points between $r/\sigma = 1.5$ and $r/\sigma = 2$ which meet the criteria for diffusivity. That is, df1 $\approx$ df2. However, most of the coefficients differ by about a factor of 100.

The average diffusion rate is decreasing with increasing rotations. The range of coefficients for $10^5$ rotations is between $10^{-10}$ and $10^{-15}$. This drop with increasing rotation number is another indication that the particle motion is still oscillatory and not diffusive. If the particles are diffusive the diffusion coefficients would settle down to values independent of the time scale. So in tracking code simulations a majority of the particles exhibit oscillatory motion up to $10^5$ rotations at different initial phase space positions.

2 $\delta f$ Simulation Results

In this section we describe particle diffusion results obtained from the $\delta f$ simulation code described in Sec. III. SSC reference parameters from Sec. A. are used with $\nu_0 = 0.285$ and
\( \Delta \nu_0 = 2.1 \times 10^{-3} \). Each beam in the simulation has \( 10^3 \) simulation particles with the initial distribution in \((x, p_x)\) phase space shown in Fig. V.

The diffusion coefficients are calculated for 100 initially uniformly distributed sample particles after 10240 rotations. As in previous sections the diffusion coefficients, \( \text{df1} \) and \( \text{df2} \), are calculated using \(|x|\). The diffusion \( D_x \) is again normalized to \( \sigma_x^2/N \). Results from the \( \delta f \) code and tracking code after 10240 rotations are shown in Fig. 16. The diffusion coefficients for the \( \delta f \) and tracking code nearly overlay each other. Both codes show oscillatory motion for 10240 rotations. These results can be compared with results where a PIC code is used in the interaction region.\(^{19}\) As stated in earlier sections the PIC code is the case of the \( \delta f \) algorithm where constant weights are used to represent the entire evolving distribution function. Figure 17 shows the diffusion coefficients obtained from the PIC code with \( 10^4 \) simulation particles initialized with equal weights and the tracking code.\(^{19}\) The noise level of the \( \delta f \) code is less than a strong-strong PIC code which shows diffusive characteristics in the tails of the distribution.

The sample particles begin to show diffusive behavior, when the number of rotations is increased. Figure 18 shows the diffusion coefficients, \( \text{df1} \) and \( \text{df2} \), calculated for 40960 rotations. Particles with \( r/\sigma > 2 \) are diffusive (\( \text{df1} \approx \text{df2} \)). The particles with \( r/\sigma < 2 \) are still somewhat oscillatory in nature. It appears that the particles in the tail of the distribution are most sensitive to either noise or collective motion in the beams.

A comparison of the \( \delta f \) and tracking code at 40960 rotations is shown in Fig. 19. The tracking and \( \delta f \) code diffusion coefficients are nearly equal to the short time scale coefficient \( \text{df1} \) with values of \( r/\sigma < 1.5 \). For the long time scale coefficient \( \text{df2} \) and \( r/\sigma > 1.5 \) the \( \delta f \) code shows more diffusive behavior. This indicates that the phenomenon which causes the diffusive motions at for large \( r/\sigma \) is most evident on time scales of 409 rotations. Diffusive motion is not evident for for particles with \( r/\sigma < 1.5 \). This indicates that the diffusion occurs on longer time scales there. It appears that the diffusion is largest for large \( r/\sigma \) and smallest
for small $r/\sigma$. This enhanced diffusion in the tails was also observed in the strong-strong PIC code with fewer rotations (Fig. 17).\textsuperscript{10} When the $\delta f$ code is run for $10^5$ rotations, all the sample particles show diffusive behavior.

V Conclusions

We have developed a new fully self-consistent $\delta f$ algorithm and one dimensional code to study the beam-beam interaction in circular colliders. It has been shown that by finding an analytic solution for the particle distribution $f_0(x, x', s)$ near the evolving particle distribution $f(x, x', s)$ the simulation of particle beams with more realistic fluctuation levels is possible. The only added computation is the calculation of the evolution of weighting functions $w_i$ for each particle. In tests of the code over short timescales the simulations show the expected beam blow-up near resonances with the rate of blow-up decreasing with the order of the resonance. Test simulations over long timescales far from resonances show that after $10^5$ rotations the two main approximations of the $\delta f$ code are still valid. The deviation from the initial Gaussian distribution is still small. After $10^8$ rotations in the $\delta f$ code the simulation particles are no longer uniformly distributed in $(x/\sigma_x, p_x/\sigma_p)$ space. In the simulations using the reference SSC parameters the amplitude of the variation in $\Delta\nu$ for small amplitude particles is approximately $\pm 3\%$ of input unperturbed tune shift $\Delta\nu_0$ throughout the run.

In studying particle diffusion away from resonances it is found that the single particle tracking code shows no diffusion of particles from the beam-beam interaction after $10^5$ rotations. The $\delta f$ code and tracking code give exactly the same diffusion coefficients up to 10240 rotations. A PIC code with $10^4$ particles and 10240 rotations already shows diffusion in the tails of the distribution. The $\delta f$ code begins to show diffusion in the tails after 40960 rotations. The coefficients near the core of the beam agree with the tracking code. All particles are diffusive after $10^5$ rotations.
Future improvements to the code will include the extension to three dimensions and a more realistic storage ring model.$^{20}$ Also, as shown in earlier sections, the perturbation equation for the $\delta f$ advance was linear in $\delta f$ (Eq. (20)). The term which is neglected is Eq. (21):

$$\delta F(x, s) \frac{\partial \delta f}{\partial x'},$$

which was assumed to be small. This term, however, can be incorporated in the $\delta f$ advance by placing it in the stationary Eq. (12):

$$\frac{\partial f_0}{\partial s} + x' \frac{\partial f_0}{\partial x} - (K(s) - F_0(s))x \frac{\partial f_0}{\partial x'} = 0$$

(75)

in the following manner:

$$\frac{\partial f_0}{\partial s} + x' \frac{\partial f_0}{\partial x} - (K(s) - F_0(s))x \frac{\partial f_0}{\partial x'} = \left< \delta F(x, s) \frac{\partial \delta f}{\partial x'} \right>$$

(76)

where $<$ $>$ refers to time average. The incorporation of this term in the stationary Eq. (12) forces the numerical advance now of $f_0(x, x', s)$. However, $f_0(x, x', s)$ is slowly varying as long as it is away from resonances, so that the equation would need to be advanced only every few thousand rotations. The term in Eq. (74) is the quasi-linear term,$^{21}$ with which the background distribution function is changing on a much larger timescale in Eq. (76). In this way we can keep track of the overall change in $f_0$ so that the order of $\delta f/f_0$ is kept small even for very long timescale simulations.

Other improvements which can be made include a higher order method of integration of the particle positions using the method of symplectic integration algorithms$^{22}$ or Lie algrebraic techniques.$^{23}$

Acknowledgments

This work was supported by the United States Department of Energy, Universities Research Association, Inc. (SSC Laboratory), and Texas National Research Laboratory Commission. We appreciate the encouragement of M. Tigner, A. Chao, and M. Syphers on this project.
References


Figure Captions

1. This shows the two components used to model the collider.

2. Uniform distribution of 1000 particles in $x, p_x$ phase space.

3. Distribution of $10^3$ simulation particles in $(x/\sigma_x, p_x/\sigma_p)$ space after 50 rotations with $\nu_0 = 0.5021$ and $\Delta \nu_0 = 2.1 \times 10^{-3}$.

4. Gaussian steady state distribution.

5. Distribution of $10^3 \delta f$ simulation particles including the particle weights in $x$ after 50 rotations with $\nu_0 = 0.5021$ and $\Delta \nu_0 = 2.1 \times 10^{-3}$.

6. The emittance $\epsilon$ of both beams for 50 rotations. One beam is at the top and the other beam is at the bottom of the figure.

7. Distribution of $10^3$ simulation particles in $(x/\sigma_x, p_x/\sigma_p)$ space after 400 rotations with $\nu_0 = 0.2521$ and $\Delta \nu_0 = 2.1 \times 10^{-3}$.

8. Distribution of $10^3 \delta f$ simulation particles including the particle weights in $x$ after 400 rotations with $\nu_0 = 0.2521$ and $\Delta \nu_0 = 2.1 \times 10^{-3}$.

9. The emittance $\epsilon$ of both beams for 400 rotations. One beam is at the top and the other beam is at the bottom of the figure.

10. The minimum and maximum perturbation values $\delta f/f_0$ for $M = 10240$ rotations, $\nu_0 = 0.285$, and $\Delta \nu_0 = 2.1^{-3}$.

11. Distribution of $10^3$ simulation particles in $(x/\sigma_x, p_x/\sigma_p)$ space after $10^5$ rotations with $\nu_0 = 0.285$ and $\Delta \nu_0 = 2.1 \times 10^{-3}$.

31
12. Distribution of $10^3 \delta f$ simulation particles including the particle weights in $x$ after $10^5$ rotations with $\nu_0 = 0.285$ and $\Delta \nu_0 = 2.1 \times 10^{-3}$.

13. Tune shift $\Delta \nu$ measured from a least squares fit to small amplitude particles $x < 0.1 \sigma_x$ (top) and all particles (bottom) for $M = 10240$ rotations.

14. $D_x$ from the tracking code with $\Delta \nu_0 = 2.1 \times 10^{-3}$ and $\nu = 0.285$ for $M = 10240$ rotations. df1 and df2 have time scales of $\Delta N_1 = 102$ and $\Delta N_2 = 1024$ rotations respectively.

15. $D_x$ from the tracking code with $\Delta \nu_0 = 2.1 \times 10^{-3}$ and $\nu = 0.285$ for $M = 40960$ rotations. df1 and df2 have time scales of $\Delta N_1 = 409$ and $\Delta N_2 = 4096$ rotations respectively.

16. $D_x$ from the $\delta f$ code with 1000 simulation particles and the tracking code for $M = 10240$ rotations. df1 and df2 have time scales of $\Delta N_1 = 102$ and $\Delta N_2 = 1024$ rotations respectively.

17. $D_x$ from tracking code and the strong-stong PIC code with $10^4$ simulation particles for $M = 10240$ rotations. df1 and df2 have time scales of $\Delta N_1 = 102$ and $\Delta N_2 = 1024$ rotations respectively.

18. $D_x$ from the $\delta f$ code with 1000 simulation particles for $M = 40960$ rotations. df1 and df2 have time scales of $\Delta N_1 = 409$ and $\Delta N_2 = 4096$ rotations respectively.

19. $D_x$ from tracking code and the $\delta f$ code for $M = 40960$ rotations. df1 and df2 have time scales of $\Delta N_1 = 409$ and $\Delta N_2 = 4096$ rotations respectively.
Fig. 1
Fig. 4
Fig. 9
Fig. 12
Fig. 13
Fig. 14
Fig. 15
Fig. 16