Collisional δf method

Yang Chen, Roscoe B. White

Princeton Plasma Physics Laboratory, P.O. Box 451, Princeton, New Jersey 08543

A general method for including various collisional effects, such as the drag and diffusion of test particles due to background plasmas, the effect of particle source and sink, and the like-particle Coulumb collisions, is presented. The marker density g is generally unknown along the particle trajectory, and its evaluation depends on the way particles are initially loaded and new particles are injected into the simulation. The method is demonstrated for the problem of the nonlinear evolution of the Toroidicity Induced Alfven eigenmode, driven by energetic α particles. The saturation amplitude is found to scale with the collision rate in a way as predicted by theory.

PACS numbers: 52.65.+z 52.35.Mw

I. Introduction

The δf method[1,2] has been widely used in particle simulation of kinetic phenomena, where the particle distribution, $f = f_0 + \delta f$, is characterized by an equilibrium distribution f_0 and a small deviation δf. The essence of this method is the following. Each particle in the simulation is assigned an initial weight which then evolves in such a way that, at any moment, the collection of simulation particles (called markers), together with their associated weights, provide a proper (Monte Carlo) representation of δf, rather than f, at that moment. Typically the number of particles used in a δf simulation can be reduced from that used in a simulation of the complete distribution function by a factor of $\delta f/f$, while achieving the same accuracy.

Despite the wide application of the δf method, there is still some conceptual difficulty concerning the interpretation of the particle weight, particularly in the case of diffusive particle motion. The δf method was initially proposed for nondiffusive, mostly Hamiltonian particle motion[1]. If the phase space is initially divided into many contiguous volume elements, these volume elements will remain contiguous, although their shapes might be continually deformed. It is then possible to define the particle weight in terms of δf at the particle position, i.e.,

$$w = \delta f/g \tag{1}$$

where g is the numerically evolved simulation particle distribution[2,3]. With this definition, the evolution equation for w can be derived straightforwardly from the kinetic equation for δf, using the fact that g is either constant (for incompressible motion) or can be advanced (for compressible motion[3]) along particle trajectories. However, this definition causes difficulty in deriving the weight equation when the particle motion is diffusive, where the "characteristic line", along which the derivative $\frac{d}{dt}w$ is to be evaluated, is stochastic. The diffusion causes initially contiguous elements in phase space to mix with each other during the subsequent evolution, and this feature makes it impossible to define the particle weight in terms of the local δf. Indeed, the systematic treatment of collisions, including both the scattering of test particles due to background plasma and, more fundamentally, binary Columb collisions as described by the nonlinear Landau collision operator, is an important challenge in δf simulation[3]. There have been a number of applications of the collisional δf method[4–7], where collisional terms in the kinetic equation are simply implemented with a Monte Carlo model in which particle velocities are changed stochastically, while no formal derivation of the weight equation is given. Since the simulation particles (the markers) do not directly represent the physical particle distribution δf, it is not clear whether the resulting algorithm correctly solves the original kinetic equation, in the limit of large particle number.

The purpose of this paper is to present a generalized collisional δf method. We abandon the definition for the particle weight given by Eq. 1, and treat w as a new dimension of the particle motion, in addition to the usual dimensions of x-v space. The simulation particles are described by a marker distribution function, F_M, in the extended phase space. This distribution satisfies a new kinetic equation, the marker kinetic equation. Eq. 1 is replaced by an equation which relates the particle weight to δf,

$$\delta f(\mathbf{x}, v, \lambda, t) = \int F_M(\mathbf{x}, v, \lambda, w, t) w dw. \qquad (2)$$

The equation for w is chosen such that the new kinetic equation, the original δf equation and Eq.2 are consistent. The approach adopted here allows a rigorous derivation of the weight equation. In particular, a nonlinear binary collisional algorithm, suitable for like particle collisions, is given. A prominent feature of the present δf method is the unconventional role of the simulation particle distribution g, which always appears in the weight equation. In some previous applications of the δf method with collisional effects[4,5], spatially uniform Maxwellian distributions are preserved by collisions, even in the presence of perturbations. In general it is impossible to know g exactly in advance. Fortunately, good approximations to g exist, as long as $\delta f / f \ll 1$.

In some important problems, such as the problem of the nonlinear interaction between the Toroidicity-Induced Alfven eigenmode (TAE) and energetic particles in a tokamak, particle birth due to fusion reactions or neutral beam injection, and particle loss (from the phase space region of interest) due to collision processes such as drag and diffusion, play an essential role in the physical process. In such cases it is necessary to introduce new particles during the simulation to keep the region of interest in phase space well populated. In this work issues such as the initialization of the newly injected particle weights, and the appropriate initial loading are discussed. It is demonstrated that collisions can provide a mechanism for renewing the particle distribution in the resonant domain, thereby allowing a steady state saturation even in the presence of a finite background damping[8]. The observed saturation amplitude also scales with the collision rate as predicted by theory.

The paper is organized as follows. In Section II we describe the collisional δf method using a model drift-kinetic equation. Relations to previous algorithms are discussed. In section III we apply the δf method to two model problems. The first is a simple one dimensional (1-D) diffusion problem, which we use to illustrate the different approaches to treating g. We then apply the δf method to the problem of nonlinear TAE evolution. Conclusions are presented in Section IV.

II. The δf method with collisions

A. Weight equation

To facilitate the discussion, we consider the following collisional drift kinetic equation for a species with distribution $f(\mathbf{x}, v, \lambda, t)$ $(\lambda = v_{\|}/v)$, in the presence of a particle source, annihilation due to charge exchange, slowing down and pitch angle scattering.

$$\frac{\partial f}{\partial t} + \mathbf{V}_H \cdot \nabla f + \dot{v}_H \frac{\partial f}{\partial v} + \dot{\lambda}_H \frac{\partial f}{\partial \lambda} - \nu_d \frac{\partial}{\partial \lambda}(1 - \lambda^2) \frac{\partial f}{\partial \lambda} - \frac{\nu}{v^2} \frac{\partial}{\partial v}[(v^3 + v_I^3) f]$$
$$= S(\mathbf{x}, v, \lambda) - \nu_a f \quad , \tag{3}$$

where $\mathbf{V}_H(\mathbf{x}, v, \lambda, t)$ is the guiding center velocity, $\dot{v}_H(\mathbf{x}, v, \lambda, t)$, $\dot{\lambda}_H(\mathbf{x}, v, \lambda, t)$ are the rate of change of velocity and pitch, respectively. The subscript H denotes that this part of the particle motion is Hamiltonian, which conserves the magnetic moment μ. The frequencies $\nu(\mathbf{x})$ and $\nu_d(\mathbf{x}, v)$ are the slowing-down rate and pitch-angle scattering rate, $S(\mathbf{x}, v, \lambda)$ is the particle source and $\nu_a(\mathbf{x}, v)$ is the annhilation rate. This equation describes a dillute species where collisions among the species can be neglected.

An external perturbation due, for example, to an electromagnetic wave, is included in $\mathbf{V}_H$, $\dot{v}_H$ and $\dot{\lambda}_H$ by adding terms $\mathbf{V}_{H1}$, $\dot{v}_{H1}$ and $\dot{\lambda}_{H1}$, i.e.,

$$\mathbf{V}_H = \mathbf{V}_{H0} + \mathbf{V}_{H1} \tag{4}$$

$$\dot{v}_H = \dot{v}_{H1} \tag{5}$$

$$\dot{\lambda}_H = \dot{\lambda}_{H0} + \dot{\lambda}_{H1}. \tag{6}$$

Since the zeroth order Hamiltonian motion conserves energy, $\dot{v}_{H0} = 0$.

Assume $f = f_0 + \delta f$, where the unperturbed distribution f_0 satisfies

$$\mathbf{V}_{H0} \cdot \nabla f_0 + \dot{v}_{H0}\frac{\partial f_0}{\partial v} + \dot{\lambda}_{H0}\frac{\partial f_0}{\partial \lambda} - \nu_d \frac{\partial}{\partial \lambda}(1-\lambda^2)\frac{\partial f_0}{\partial \lambda} - \frac{\nu}{v^2}\frac{\partial}{\partial v}[(v^3 + v_I^3)f_0]$$
$$= S(\mathbf{x}, v, \lambda) - \nu_a f_0 \ , \tag{7}$$

and δf satisfies

$$\frac{D}{Dt}\delta f = -\mathbf{V}_{H1} \cdot \nabla f_0 - \dot{v}_{H1}\frac{\partial f_0}{\partial v} - \dot{\lambda}_{H1}\frac{\partial f_0}{\partial \lambda} - \nu_a \delta f \tag{8}$$

where the notation $\frac{D}{Dt}(f)$ denotes, for arbitrary f,

$$\frac{D}{Dt}f = \frac{\partial f}{\partial t} + \mathbf{V}_H \cdot \nabla f + \dot{v}_H \frac{\partial f}{\partial v} + \dot{\lambda}_H \frac{\partial f}{\partial \lambda} - \nu_d \frac{\partial}{\partial \lambda}(1-\lambda^2)\frac{\partial f}{\partial \lambda} - \frac{\nu}{v^2}\frac{\partial}{\partial v}[(v^3 + v_I^3)f] \tag{9}$$

Particle trajectories in the phase space $(\mathbf{x}, v, \lambda)$ can be identified from Eq. 8 or Eq. 3 as,

$$\frac{d\mathbf{x}}{dt} = \mathbf{V}_H \tag{10}$$

$$\frac{dv}{dt} = \dot{v}_H - \nu(v + \frac{v_I^3}{v^2}) \tag{11}$$

$$\frac{d\lambda}{dt} = \dot{\lambda}_H - 2\nu_d\lambda + (2\nu_d(1-\lambda^2))^{1/2}\Gamma(t) \tag{12}$$

where $\Gamma(t)$ is (Gaussian) white noise with $<\Gamma(t)>= 0$ and $<\Gamma(t)\Gamma(t+\tau)>= \delta(\tau)$.

The markers in the simulation follow the trajectories defined by Eq. 10-12. In addition, they have a new dimension, the weight w, which evolves according to,

$$\frac{dw}{dt} = \dot{w}(\mathbf{x}, v, \lambda, w, t) \tag{13}$$

where the function $\dot{w}(\mathbf{x}, v, \lambda, w, t)$ is to be determined.

To derive the weight equation it is most convenient to introduce a marker distribution function in the extended phase space $(\mathbf{x}, v, \lambda, w)$, $F_M(\mathbf{x}, v, \lambda, w, t)$, which, according to the equivalence between Langevin equations and the Fokker-Planck equation, satisfies the following marker kinetic equation,

$$\frac{D}{Dt} F_M + \frac{\partial}{\partial w}(\dot{w} F_M) = S_M(\mathbf{x}, v, \lambda, w, t) \ , \tag{14}$$

where in anticipation of possible marker loss in the simulation, we have allowed a source S_M for the markers, to provide, in addition to the initial loading, further control of the marker population. Note that the left-hand side of Eq. 14 is not in the standard Fokker-Planck form. It can be converted into the standard form using the fact that the Hamiltonian part of the motion is incompressible.

In the simulation F_M is represented as

$$\tilde{F}_M(\mathbf{x}, v, \lambda, w, t) = \sum_j \delta(\mathbf{x} - \mathbf{x}_j(t)) \delta(v - v_j(t)) \delta(\lambda - \lambda_j(t)) \delta(w - w_j(t)) \ , \tag{15}$$

and δf represented as

$$\tilde{\delta f}(\mathbf{x}, v, \lambda, t) = \sum_j w_j \delta(\mathbf{x} - \mathbf{x}_j(t)) \delta(v - v_j(t)) \delta(\lambda - \lambda_j(t)) \ . \tag{16}$$

This suggests that the smooth functions F_M and δf be related by

$$\delta f(\mathbf{x}, v, \lambda, t) = \int F_M(\mathbf{x}, v, \lambda, w, t) w dw \ . \tag{17}$$

The weight change rate, $\dot{w}$, can be determined by requiring that Eq. 8, Eq. 14 and Eq. 17 are consistent. Multiply Eq. 14 by w and then integrate over w, and compare the resulting equation to Eq. 8. Noticing that $\mathbf{V}_H$, $\dot{v}_H$, $\dot{\lambda}_H$ are independent of w, we find,

$$\int \dot{w} F_M dw = -\int S_M w dw - \mathbf{V}_{H1} \cdot \nabla f_0 - \dot{v}_{H1} \frac{\partial f_0}{\partial v} - \dot{\lambda}_{H1} \frac{\partial f_0}{\partial \lambda} - \nu_a \int w F_M dw. \qquad (18)$$

To get an explicit weight equation we assume $\dot{w}$ to be independent of w, therefore $\dot{w}$ can be taken out of the integral on the left-hand side. Defining the integrated marker distribution function in the original phase space,

$$g(\mathbf{x}, v, \lambda, t) = \int F_M(\mathbf{x}, v, \lambda, w, t) dw \qquad (19)$$

we find the weight evolution equation for the δf method to be

$$\dot{w} = \frac{1}{g} \left[-\int S_M w dw - \mathbf{V}_{H1} \cdot \nabla f_0 - \dot{v}_{H1} \frac{\partial f_0}{\partial v} - \dot{\lambda}_{H1} \frac{\partial f_0}{\partial \lambda} - \nu_a \int w F_M dw \right] . \qquad (20)$$

This is indeed independent of w. The functional dependence of $\dot{w}$ on the distribution F_M can be compared to, for example, the dependence of the electrostatic force on the particle distribution in a 1-D particle simulation, as described by the Poisson equation. In general, the annihilation term can be replaced by any term that might involve integrals of δf and hence F_M, such as that resulting from linearizing the full collision operator. That g appears in the denominator of the weight equation is familiar in δf algorithms. Intuitively, to correctly account for the increment of δf at $(\mathbf{x}, v, \lambda)$, the total increment, the source term in Eq. 8, should be evenly distributed to all the markers at that point.

The distribution g satisfies

$$\frac{D}{Dt} g = \int S_M dw \ . \qquad (21)$$

It is the distribution F_M that is being sampled by the markers, and the markers are to be added to or removed from the phase space according to the source S_M, which can be chosen in any convenient form. In particular, new simulation particles do not have to be created according to the physical particle birth rate S, nor do they have to be removed according to the annihilation rate of physical particles. This feature makes it possible to arrange the marker density g in favor of phase space domains of interest, such as the resonance domains in wave-particle interaction problems. The initial distribution, $F_M(\mathbf{x}, v, \lambda, w, t = 0)$, according to which the markers are to be loaded initially, can also be arbitrary as long as the initial condition for δf is satisfied,

$$\int wF_M(\mathbf{x}, v, \lambda, w, 0)dw = \delta f((\mathbf{x}, v, \lambda, t = 0). \tag{22}$$

Boundaries should be treated in a manner consistent with the physical boundary conditions. For example, if physical particles are considered to be lost at the boundary $r = a$, then markers hitting this boundary should be removed from the simulation. The evolution of weight is unbounded. Serious questions concerning the well-posedness of Eq. 14 might arise when particles are absorbed at the boundary of a dimension along which the motion is diffusive, as in this case $g = 0$ (as well as $F_M = 0$ and $f_0 = 0$) must hold at that boundary, causing a formal singularity in Eq. 20 and Eq. 14. To avoid such a singularity, one can replace g in Eq. 20 by $g + \delta$. It can be argued that the simulated δf approaches the exact solution for small δ, except at a small boundary layer. Approximations to g will be discussed later in this paper.

B. Nonlinear collisional algorithm

The collisional algorithm just described is restricted to a linearized collision operator when applied to like particle collisions. Consider the Landau collision operator for a one component plasma $C(f,f)$ with the operator C defined as

$$C(f_1,f_2) = \Gamma\frac{\partial}{\partial\mathbf{v}}\cdot\int d\mathbf{v}'\mathsf{U}(\mathbf{u})\cdot(\frac{\partial}{\partial\mathbf{v}}-\frac{\partial}{\partial\mathbf{v}'})f_1(\mathbf{x},\mathbf{v}',t)f_2(\mathbf{x},\mathbf{v},t) \tag{23}$$

with $\Gamma = q^4 ln\Lambda/4\pi\varepsilon_0^2 m^2$, $\mathbf{u}=\mathbf{v}-\mathbf{v}'$ and $\mathsf{U}(\mathbf{u}) = (u^2\mathsf{I}-\mathbf{uu})/u^3$.

The nonlinear operator $C(f,f)$ can be expanded as

$$C(f,f)-C(f_0,f_0) = C(f_0,\delta f)+C(\delta f,f_0)+C(\delta f,\delta f). \tag{24}$$

The first term on the right hand side represents drag and diffusion of test particles (represented by δf) due to the zero order distribution f_0. If the nonlinear term $C(\delta f,\delta f)$ is neglected, then $C(f,f)$ can be treated using the previous approach, with $C(\delta f,f_0)$, which involves only integrals of δf, included in the weight equation as a generalization of the annihilation term in Eq. 8. The nonlinear term $C(\delta f,\delta f)$ can not be treated similarly because it involves differentials of δf. As we shall show, the binary approach naturally takes care of this term.

In a binary collisional algorithm for full-f particle simulations[9–11], the collision operator is treated as follows:

1. The simulation system is divided into a number of spatial cells with a size such that plasma properties across each cell do not vary substantially.

2. Particles in each cell are paired in a random way.

3. Small angle collisions are performed pairwise. Let $\mathbf{v}$ and $\mathbf{v}'$ be the velocity of the particles in the pair. After collision, the change in relative velocity $\mathbf{u} = \mathbf{v} - \mathbf{v}'$ is given by

$$\triangle u_x = \frac{u_x}{u_\perp} u_z \; sin\Theta \; cos\Phi - \frac{u_y}{u_\perp} u \; sin\Theta \; sin\Phi - u_x(1 - cos\Theta) \tag{25}$$

$$\triangle u_y = \frac{u_y}{u_\perp} u_z \; sin\Theta \; cos\Phi + \frac{u_x}{u_\perp} u \; sin\Theta \; sin\Phi - u_y(1 - cos\Theta) \tag{26}$$

$$\triangle u_z = -u_\perp \; sin\Theta \; cos\Phi - u_z(1 - cos\Theta) \tag{27}$$

where $u_\perp = (u_x^2 + u_y^2)^{1/2} \neq 0$, Θ is a Gaussian random variable with mean 0 and variance $16\Gamma\triangle t/u^3$, and Φ is a uniform random value in $(0, 2\pi)$. The changes to $\mathbf{v}$ and $\mathbf{v}'$ are given accordingly by,

$$\triangle \mathbf{v} = \frac{1}{2}\triangle \mathbf{u} \tag{28}$$

$$\triangle \mathbf{v}' = -\frac{1}{2}\triangle \mathbf{u} \tag{29}$$

The small angle collision in step 3 conserves the energy and momentum pairwise, and particle number is conserved automatically in a full-f simulation. Over many time steps, the random choice of pairing for a particular test particle in step 2 in effect performs the integral over field particles in Eq. 23.

This binary algorithm can be adopted for δf simulation. In particular, particles inside a spatial cell are randomly paired regardless of their weights. We must consider how to include collisional effects in the weight equation. It is clear that the total perturbed particle number $\delta N = \sum_j w_j$, kinetic energy $\delta K = \sum_j w_j \varepsilon_j$ ($\varepsilon_j = 1/2mv_j^2$) and momentum $\delta \mathbf{M} = \sum_j w_j m \mathbf{v}_j$ are not conserved if collision effects are included merely through such binary collisions, with the equation for particle weight unchanged. It is also impossible to change the particle weights such that these quantities are conserved pairwise, although it is possible to pairwise conserve the particle number and energy. In any case, simulation schemes which solely enforce these conservation properties, while useful in some applications, do not treat the collision operator faithfully, regardless of the particle number used.

To find the exact weight evolution equation for the binary collisional algorithm, let us consider how to represent the binary collision in the kinetic equation for the marker distribution $F_M(\mathbf{x}, \mathbf{v}, w, t)$. Since we randomly pair the particles in a spatial cell, regardless of their weights, and particle velocities change in the same manner as in full-f simulation, the corresponding collision operator for F_M is

$$C_M(F_M, F_M) = \Gamma \frac{\partial}{\partial \mathbf{v}} \cdot \int dw' \int d\mathbf{v}' \mathsf{U}(\mathbf{u}) \cdot (\frac{\partial}{\partial \mathbf{v}} - \frac{\partial}{\partial \mathbf{v}'}) F_M(\mathbf{x}, \mathbf{v}', w', t) F_M(\mathbf{x}, \mathbf{v}, w, t) \quad (30)$$

This collisional operator for F_M is equivalent to the following in the δf equation,

$$\int dw \; w C_M(F_M, F_M) = C(g, \delta f) \quad (31)$$

where we have used $g = \int F_M dw$ and $\delta f = \int w F_M dw$. If we choose the initial loading and marker source S_M such that $g = f$, then this collision term covers the first and third term on the right hand side of Eq. 24. Hence the weight equation should include a term $\dot{w}_c$,

$$\dot{w}_c = \frac{1}{g} C(\delta f, f_0) \tag{32}$$

Interestingly, this is the same term we would add to the weight equation if we had used the linearized collision operator using the previous algorithm. This might be expected since the binary collision algorithm adopted here only provides an alternative way of treating the random marker trajectory in $(\mathbf{x}, \mathbf{v})$ space, except that it is fully nonlinear. A more thorough binary approach would have treated the particle weights the same way as the velocities, i.e., making all the collisional modifications to the weights at the binary collision step. Such an algorithm has been considered previously for the linearized collision operator, and is found to be impractical for δf simulation, because the number of markers increases rapidly as a function of time[5].

We emphasize that to achieve a fully nonlinear collisional algorithm in the δf method, we lose the freedom of choosing an arbitrary marker distribution g, and are forced to load and evolve the simulation particle population according to the physical distribution f. A vectorization scheme for the collisional steps 1-3 has been previously discussed[10] for a full-f gyrokinetic simulation. A remark on the evaluation of $C(\delta f, f_0)$ is in order. Consider a particular marker with index m. Since $C(\delta f, f_0)$ can be converted to an integral of $\delta f(\mathbf{x}_m, \mathbf{v})$ over $\mathbf{v}$ at the marker position $\mathbf{x}_m$, it can in principle be expressed as a sum over all the markers in the spatial cell to which marker m belongs, using Eq. 16. However, due to the $\mathsf{U}(\mathbf{u})$ (singular when $\mathbf{u} = 0$) dependence of $C(\delta f, f_0)$, contributions from those markers with velocities close to $\mathbf{v}_m$ will cause numerical difficulties. One solution is to discard the contributions from those particles with velocity $\mathbf{v}'$ close to $\mathbf{v}_m$, $|\mathbf{v}' - \mathbf{v}_m| < \varepsilon$, for some ε. Another possibility is to expand δf in terms of some basis functions, with the expansion coefficients evaluated by summing over markers, and the contribution of each basis function to $C(\delta f, f_0)$ calculated in advance. It is also possible in some applications to replace $C(\delta f, f_0)$ by a simpler term which is not singular, yet ensures some desired conservation properties[5,4].

C. Relation to other algorithms

If we take $S_M = 0$ and $\nu_a = 0$, Eq. 20 takes the form similar to Eq. 11 in a previous work[2]. However, Eq. 1, which is crucial for the derivation of the weight equation in nondiffusive motion, does not hold here. Due to the diffusive motion, particles located in any (arbitrarily small) region of $(\mathbf{x}, v, \lambda)$ will have different weights even if they all start with $w = 0$. In the case of Hamiltonian motion or nondiffusive motion in general, if we start all the markers with $w = 0$, then the solution of Eq. 14 has the form $F_M = g(\mathbf{x}, v, \lambda)\delta(w - w(\mathbf{x}, v, \lambda))$, hence $\delta f(\mathbf{x}, v, \lambda) = g(\mathbf{x}, v, \lambda) w(\mathbf{x}, v, \lambda)$ and we recover the previous result Eq. 1.

When only the charge exchange term $-\nu_a \delta f$, or a Krook collision model is retained in Eq. 3, it's effect can be accounted for by adding a term $-\nu_a w$ in the weight equation[3]. This makes $\dot{w}$ dependent on w, but Eq. 18 is still satisfied. Alternatively, particle annihilation can be taken into account by removing randomly selected markers from the simulation using the Monte Carlo technique, with the weight equation unchanged. This amounts to including a term $-\nu_a F_M$ in S_M. In principle one can include any of the derivative terms except the first term in the operator $\frac{D}{Dt}$ in $\dot{w}$ and let the rest of $\frac{D}{Dt}$ define the characteristics of the markers. In the extreme case where only the $\frac{\partial}{\partial t}$ term is used to define the characteristics, one arrives at a finite difference algorithm for the partial differential equation Eq. 8. Since integrals can be evaluated using Monte Carlo methods more accurately than derivatives, generally only non-differential terms are retained for the weight equation.

If Eq. 21, with some choice of S_M, allows an explicit solution, for example, with $S_M = 0$, $g = 1$ when the motion is Hamiltonian or only pitch angle scattering is considered, or g is given by a spatially uniform Maxwellian distribution[4,5], that distribution can be chosen to load the markers, and the evaluation of g at the marker position is avoided. However, this is generally impossible when the drag or diffusion coefficient is velocity dependent, as the case considered here.

D. Evaluation of g

So far in the applications of the δf method the evaluation of g at the marker position has been avoided, by loading the markers according to a known explicit solution of the kinetic equations[4,5], or by using the fact that in nondiffusive motion g or equivalently, the volume element, can be advanced along the particle trajectory[3]. Estimating g from the marker density Eq. 15 is statistically demanding, and should be avoided whenever possible. One solution is to choose $F_M(0) = f(0)\delta(w)$ and $S_M = S\delta(w) - \nu_a F_M$. Eq. 21 and Eq. 3 are then identical and we have $g = f$. One can then estimate g through δf according to $g = f_0 + \delta f$, with much reduced noise. Often it is easy to choose S_M and the initial loading $F_M(t = 0)$ such that at a later time $g(t)$ is only slightly perturbed from $g(t = 0)$, in the same spirit as f is only slightly perturbed from f_0. One can then approximate g by $g(0)$, although always evaluated at the marker position. We envision this as the most practical approach, and will use it for the TAE problem in Section III. However, when applied to steady state problems, such as the evaluation of the steady state plasma current[6], this approximation might cause a secular change in the total particle number, as will be illustrated by an example in Section III.

III. Examples

A. 1-D diffusion

Let us solve the following 1-D problem using the δf method,

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}(V(x)f) - D\frac{\partial^2 f}{\partial x^2} = 0 \tag{33}$$

$$f(x, t = 0) = f_0(x)$$

with periodic boundary conditions at $x = 0$ and $x = 2\pi$. Here $V(x, t) = sin(x) + V_1(x, t)$ with $V_1(x, t) = \varepsilon cos(x - \omega t)$, D a constant, and $f_0(x) = exp(-\frac{1}{D}cos(x))$ is the steady state solution of Eq. 33 with $\varepsilon = 0$. Let $\delta f = f - f_0$. According to the previous discussion, δf can be simulated using markers whose position (the only coordinate for this problem) and weight advance according to,

$$\frac{dx}{dt} = V(x) + (2D)^{1/2}\Gamma(t) \tag{34}$$

$$\frac{dw}{dt} = -\frac{1}{g}\frac{\partial}{\partial x}(\varepsilon cos(x - \omega t)f_0) \tag{35}$$

Where we have chosen $S_M = 0$. We load $F_M(0) = f_0\delta(w)$, thus $g = f$. We compare the three schemes for evaluating g:

1. Use the approximation $g \approx f_0$.

2. Evaluate g from Eq. 15, by dividing the domain $[0, 2\pi]$ with a mesh with a spacing of equal length, and counting the number of particles in the grid (with proper normalization).

3. Estimate g from $f = f_0 + \delta f$.

Take $D = 0.3$, $\varepsilon = 0.1$, $\omega = 5$, with 10000 particles. The simulated δf at $t = 10$ is shown in Fig 1 using scheme 1, together with the exact solution (computed using a finite difference method). The other two schemes give comparable results. For $\omega = 0.$, scheme 2 and 3 still provide an accurate estimate for δf. Scheme 1, however, causes the total particle number to increase secularly with time, and by the time $t = 10$ the simulated δf is totally invalid. This can be explained as follows.

Without particle source and sink, particle number should be conserved, or,

$$\frac{\partial}{\partial t}\int \delta f dx = 0 \ . \tag{36}$$

In the δf method this conservation is not enforced and is subject to noise. When approximations to g are used, secular change to total particle number might occur. By replacing g in the weight equation with another function $g^{'}$, we are solving the following equation,

$$\frac{\partial \delta f}{\partial t} + \frac{\partial}{\partial x}(V(x)\delta f) - D\frac{\partial^2 \delta f}{\partial x^2} = -\frac{g}{g'}\frac{\partial}{\partial x}(\varepsilon V_1 f_0) \tag{37}$$

therefore the total (perturbed) particle number changes according to

$$\frac{d}{dt}\int \delta f dx = -\int \frac{g}{g'}\frac{\partial}{\partial x}(\varepsilon V_1 f_0)dx \tag{38}$$

For $\omega = 0$, eventually g will approach a steady state, and the right-hand side of Eq. 38 will approach a finite value, causing secular change in particle number, unless $g^{'} = g$. for $\omega > 0$, this expression oscillates and only a small secular change in particle number occurs.

If one is only interested in a quantity involving integrals of δf, the particle number used in the simulation can be reduced significantly. As an example, consider the quantity

$$I(t) = -\int V_1(x,t) sin(x - \omega t + \frac{\pi}{4}) \delta f dx. \tag{39}$$

With 300 particles, the computed I is shown in Fig 2 for $\omega = 0$.

B. TAE problem

Consider the evolution of a Toroidicity-Induced-Alfvén-Eigenmode (TAE) in a tokamak, excited by alpha particles, whose distribution satisfies the drift kinetic equation Eq. 3. The collisionless guiding center motion is the same as that described in Wu et.al.[12]. The unperturbed magnetic field is $\mathbf{B} = g\nabla\phi + I\nabla\theta$, where g and I are related to the poloidal and toroidal current. The guiding-center Hamiltonian is

$$\mathcal{H} = \frac{1}{2}\rho_\parallel^2 B^2 + \mu B + \Phi \tag{40}$$

with four Hamiltonian variables

$$\phi,\ \theta,\ P_\phi = g\rho_c - \psi_p,\ P_\theta = I\rho_c + \psi_t \tag{41}$$

where $\rho_c = \rho_\parallel + \tilde{\alpha}, \rho_\parallel = v_\parallel / B$, ψ_t is the toroidal flux with $d\psi_t/d\psi_p = q$. Φ is the mode electric potential and $\tilde{\alpha}$ gives the magnetic perturbation of the mode through $\delta\mathbf{B} = \nabla\times\tilde{\alpha}\mathbf{B}$. The Hamiltonian motion conserves μ. $\mathbf{V}_H$, $\dot{v}_H$ and $\dot{\lambda}_H$ in Eq.3 are implicitly given by

$$\dot{P}_\phi = -\frac{\partial\mathcal{H}}{\partial\phi}, \qquad \dot{\phi} = \frac{\partial\mathcal{H}}{\partial P_\phi} \tag{42}$$

$$\dot{P}_\theta = -\frac{\partial \mathcal{H}}{\partial \theta}, \qquad \dot{\theta} = \frac{\partial \mathcal{H}}{\partial P_\theta} \tag{43}$$

When collisions are included, each particle pushing step consists of a collisionless step which advances the particle following the Hamiltonian equations, and a collision step which performs the velocity slowing down and random scattering of the pitch angle.

The MHD displacement of the TAE is assumed to be of the form

$$\xi(\mathbf{x}, t) = A(t)\eta(\mathbf{x}) sin(\omega t + \alpha(t)) \tag{44}$$

with eigenfrequency ω, mode structure $\eta(\mathbf{x})$. The amplitude $A(t)$, and phase $\alpha(t)$ are assumed to be slowly-varying. Starting from the linearized kinetic-MHD equation[13], one can derive the following equations for mode amplitude and phase,

$$\frac{dA}{dt} = < \int q\mathbf{V}_H \cdot \mathbf{E}\delta f d\tau > /(\omega^2 A) - \gamma_d A \tag{45}$$

$$\frac{d\alpha}{dt} = - < \int q\mathbf{V}_H \cdot \frac{1}{\omega}\frac{\partial \mathbf{E}}{\partial t}\delta f d\tau > /(\omega^2 A^2) \tag{46}$$

where $d\tau = d^3\mathbf{r}d^3\mathbf{v}$, $\mathbf{E} = -\partial\xi/\partial t \times \mathbf{B}$ the mode electric field, $\gamma = \frac{dA}{dt}/A$ the growth rate, γ_d is the background damping rate. The damping mechanism might cause a phase shift term in Eq. 46 as well, but that is neglected.

The nonlinear evolution problem involves self-consistently solving the drift-kinetic equation Eq. 3 and Eqs. 45-46 for mode amplitude and phase, starting from a small initial amplitude and the unperturbed distribution f_0. This problem is important for predicting the TAE mode activity in an ignited tokamak. The importance of collisions in such problems[8] is that they provide a mechanism for refreshing the particle distribution in the resonance domain, therefore allowing a steady state solution even when a finite background damping is present.

Let γ_L denote the linear growth rate of the mode without damping and collisions, and ν_{eff} denote the rate at which a resonant particle would move out of resonance due to diffusion[8]. It has been shown that pitch angle scattering dominates this process, and $\nu_{eff} \approx \nu_d(\omega/\omega_b)^2$, where ω_b is the bounce frequency of particles trapped in the wave. Generally the nonlinear response of the mode amplitude is very complicated, depending on the three parameters γ_L, γ_d and ν_{eff}. However, when damping is weak, $\nu_{eff} > \gamma_d$, steady state response is predicted, with the steady state amplitude scaling as

$$A \sim (\gamma_L \nu_d \omega^2/\gamma_d)^{2/3} . \tag{47}$$

Consider a tokamak with circular cross section. The equilibrium magnetic field is defined by $g = 1$ and $I = r^2/q$ with $q(r)$ the safety factor[14]. Assume $\nu_a = 0$, $\nu = const$ and $\nu_d = cv_I^3/v^3$ with c a constant of order unity. We take $c = 1$. Particles moving beyond the $r = a$ surface are considered lost. We also restrict the simulation to a velocity domain (v_b, v_t), i.e., a particle is also considered to be lost when its velocity moves out of this domain. This is acceptable if (v_b, v_t) includes the most important resonance velocities. We now specify S_M and $F_M(t = 0)$, i.e., the source that controls the injection of new markers and the initial loading. The important domains in this problem are those where resonance wave-particle interaction is strong, but the criterion is very complicated when the orbits have finite widths due to drift, and we simply choose

$$S_M(\mathbf{x}, v, \lambda, t) = \delta(v - v_t)\delta(w) \tag{48}$$

which specifies that markers are constantly created with zero weight at velocity $v = v_t$, uniform in space and pitch. To be consistent with the exact Hamiltonian guiding center equations, the steady state marker density g_0 and particle distribution f_0 should be computed by solving Eq. 21 (with $\frac{\partial}{\partial t} = 0$) and Eq. 7 numerically. For simplicity we use the following approximations

$$f_0(r, v) = e^{-r^2/\triangle^2}/(v^3 + v_I^3) \tag{49}$$

$$g_0(r, v) = 1/(v^3 + v_I^3) \tag{50}$$

with $\triangle = a/3$. The equilibrium is maintained by the balance of particle source and particle slowing down. Markers are initially loaded according to $F_M(t = 0) = g_0\delta(w)$, and we use $g \approx g_0$ in Eq. 20 for advancing particle weights.

In the following simulation we choose parameters as: on-axis $B_0 = 6T$, major radius $R_0 = 8m$, inverse aspect ratio $\varepsilon = 0.375$. $V_I = 0.35v_0$, with v_0 the birth velocity of α particles. The velocity domain $(v_b, v_t) = (0.32v_0, v_0)$. The $n = 3$ mode is considered, with mode angular frequency $\omega = 1.12 \times 10^{-3}\Omega$, where Ω is the on-axis gyro-frequency of alpha particles. Mode structure and the $q(r)$ profile are shown in Fig 3. The mode structure and frequency are computed by the NOVA-K code[15], using the corresponding low-β equilibrium.

The time step chosen was $\triangle t = 0.1T$ where $T = 2\pi R_0/v_0$, the on-axis transit time. Initially 93000 particles were loaded. Particles are injected into the simulation periodically (according to Eq. 14), typically every hundred steps. During the simulation the total particle number drops slightly, due to particle loss induced by the mode. For the cases considered here, the final number is above 90000, which is consistent with the approximation $g \approx g_0$. For $\gamma_L = 0.015\omega$, $\gamma_d = 0.003\omega$, Fig 4 shows the simulated amplitudes for different collision rates. Our normalization is such that the actual peak value of $\delta \mathbf{B}_r/B_0$ is about 10 times the amplitude shown. The "natural" saturation corresponds to $\gamma_d = 0$, $\nu = 0$, which is computed by turning off the collision step and the particle injection. In this case we have the exact solution for marker density, $g(\mathbf{x}(t), v(t), \lambda(t), t) = g(\mathbf{x}(0), v(0), \lambda(0), 0)$, but the approximation $g \approx g_0$ is still used.

For given γ_d and γ_L, saturation amplitudes increase with the collision frequency, and amplitudes both above ($\nu = 20ms^{-1}$) and below ($\nu = 150ms^{-1}$) the natural level are observed. The saturation amplitude can not be determined accurately, as is evident from Fig 4, therefore it's difficult to perform a detailed study on the $A \sim \nu$ scaling. If saturation amplitudes are taken to be $A_{\nu=0.5} = 7 \times 10^{-5}$ and $A_{\nu=2.0} = 1.95 \times 10^{-4}$, as indicated in Fig 4, we have

$$\frac{A_{\nu=2.0}}{A_{\nu=0.5}} = 2.8$$

which agrees well with that predicted by Eq. 47,

$$(\frac{\nu_1}{\nu_2})^{2/3} = 2.6$$

IV. Conclusion

In this paper we presented a general method of including various collisional effects in the δf particle simulation. The method features a new interpretation of the particle weight, the unconventional role of the marker density g, the capability of treating both drag and diffusion of test particles due to background plasmas, the capability of treating fully nonlinear binary collisions, and the capability of including the effect of particle source and sink. We demonstrated this generalized method for the important problem of nonlinear TAE evolution.

ACKNOWLEDGMENTS

The authors are grateful to Dr. G. Y. Fu for helpful discussions. This work was supported by the U.S. Department of Energy under contract number DE-AC02-76-CHO3073.

1. M. Kotschenruether, Proceedings of the 14th International Conference on the Numerical Simulation of Plasmas(Office of Naval Research, Arlington, VA), Paper PT20 (1991).

2. S. Parker and W. W. Lee, Phys. Fluids B **5**, 77 (1993).

3. G. Hu and J. Krommes, Phys. Plasmas **1**, 863 (1994).

4. X. Xu and M.N.Rosenbluth, Phys. Fluids B **3(3)**, 627 (1991).

5. A. M. Dimits and B. Cohen, Phys. Rev. **49**, 709 (1994).

6. M.Sasinowski and A. Boozer, Phys. Plasmas **2**, 613 (1995).

7. Z. Lin, W. Tang, and W. Lee, Phys. Rev. Lett. **78**, 456 (1997).

8. H. L. Berk, B. N. Breizman, and H. Ye, Phys. Rev. Lett. **68**, 3563 (1992).

9. T. Takizuka and H. Abe, J. Comput. Phys. **25**, 205 (1977).

10. S. Ma, R. Sydora, and J. Dawson, Computer Physics Communications **77**, 190 (1993).

11. R. Procassini, C. Birdsall, and B. Cohen, Nucl. Fusion **30**, 2329 (1990).

12. Y. Wu, R. B. White, Y.Chen, and M.N.Rosenbluth, Phys. Plasmas **2(12)**, 4555 (1995).

13. W. Park, S. Parker, H. Biglari, M. Chance, L. Chen, C. Z. Cheng, T. S. Hahm, W. W. Lee, R. Kulsrud, D. Monticello, L. Sugiyma, and R. B. White, Phys. Fluids B **4**, 2033 (1992).

14. R. B. White and M. S. Chance, Phys. Fluids **27**, 2455 (1984).

15. C. Z. Cheng, Phys. Reports **211**, 1 (1992).

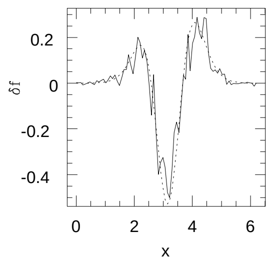

Fig. 1

Fig. 1. Simulated δf vs. exact solution, dash line the exact value

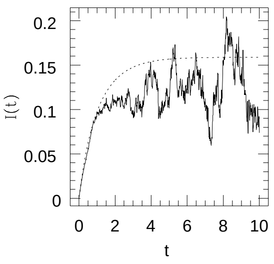

Fig. 2

Fig. 2. I [Eq. 39] from simulation with 300 particles, dash line the exact value

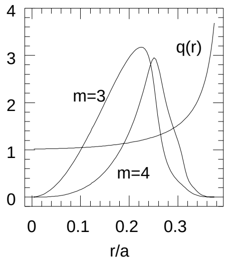

Fig. 3

Fig. 3. Mode structure for $n = 3$ and q-profile

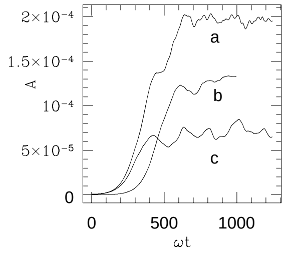

Fig 4

Fig. 4. Mode evolution for a: $\gamma_d = 0.003\omega$, $\nu = 20ms^{-1}$. b: $\gamma_d = 0$, $\nu = 0$. c: $\gamma_d = 0.003\omega$, $\nu = 150ms^{-1}$. Natural saturation corresponds to b.