JOURNAL OF COMPUTATIONAL PHYSICS 112, 226–233 (1994)

Kinetic Simulation of a Time-Dependent Two-Dimensional Plasma

W. N. G. HITCHON AND E. R. KEITER

Engineering Research Center for Plasma-Aided Manufacturing, University of Wisconsin-Madison, 1415 Johnson Dr., Madison, Wisconsin 53706

Received March 12, 1993; revised November 2, 1993

The distribution function of ions is calculated in a two-dimensional plasma with a rapidly expanding sheath, self-consistently with the electrostatic potential, Φ. The numerical procedure consists of a direct solution of an integral form of the kinetic equation. This solution relies on the use of a simple form for the Green's function which describes the time-evolution of the distribution, which has previously been used in one spatial dimension and is here extended to two dimensions. The electron density n_e is assumed to be described by the Boltzmann relation, $n_e = n_0 \exp(e\Phi/\kappa T_e)$, allowing Poisson's equation to be solved for Φ self-consistently with the ion density. This procedure is applied to describe the plasma surrounding a "target" to which is rapidly applied a large negative potential, as occurs in plasma source ion implantation (PSII). The ion distribution striking the target is calculated to allow determination of the dose and depth profile.

INTRODUCTION

Accurate simulation of many plasmas requires that a model be employed which is both kinetic and multi-dimensional. Few previous calculations of this type have been published, and these have typically been "particle simulations" [1–3]. There is no published work to date that is kinetic and multi-dimensional and that has been applied to PSII. In this paper we describe a kinetic simulation of plasma ions which is based on direct calculation of the distribution function. The distribution is found self-consistently with the electrostatic potential, Φ, in two spatial coordinates (x, y) or (r, z) as it evolves in time, t.

The distribution is advanced in time using a numerical Green's function (or propagator) which was used previously in one spatial dimension (see, e.g., [4, 5]). The advantages of the calculation are that it is accurate (realistically incorporating the actual behavior), efficient, and very easy to implement.

A calculation of the distribution such as this represents the distribution in terms of its values at a finite number of mesh points. It is necessarily limited in resolution by the mesh. A particle simulation is not directly limited to the resolution of the mesh (although Poisson's equation must be employed in particle simulation, and this is solved on a mesh). On the other hand, the distribution which we calculate does extend throughout all of the mesh (at least where the density is not truly negligible) at all times and is "infinitely" divisible. Thus it is possible to achieve excellent resolution of regions with few particles such as the tail of the distribution and the sheaths [5].

Eastwood [2] has presented an approach, ephemeral particle in cell (EPIC) that, like the propagator method, uses the method of characteristics. The main difference between the two methods is in how the method of characteristics is implemented. Our method uses a fixed (usually uniform) mesh spacing for the "initial" mesh in a time step and moves these initial cells, to be remapped onto the "old" mesh at the end of the time step. EPIC uses a uniform spacing for its final locations, rather than the initial ones, and then "looks back" along the characteristics to find the initial locations at the start of the time step. It is vital to use the former approach in order to be able to conserve numbers and energy exactly and locally in phase space. EPIC has also only been applied to simple fluid calculations, so the details of how to use it in kinetic calculations have not been worked out.

In summary, this work involves developing a kinetic plasma model which is an alternative to particle simulation with somewhat different strengths and weaknesses. The present application is the first use of the method employed here in more than one spatial dimension.

What follows includes a detailed explanation of the method used, which consists of two main components, calculating the ion distribution function and calculating the electrostatic potential. Section II deals with the numerics of the ion behavior, which has been subdivided into ballistic motion and collisions. Section III discusses the electrostatic calculation, which also contains the assumptions relevant to the electrons. Finally, in Section IV, the method described is applied to a plasma with a rapidly expanding sheath, such as is used in PSII. The results of a simulation are presented and discussed.

II. TWO-DIMENSIONAL KINETIC SIMULATION

The numerical procedure is now described in detail. Formally, the Boltzmann equation can be rewritten as an

0021-9991/94 $6.00

integral equation for the "new" distribution function, $f(\mathbf{x}, \mathbf{v}, t+\Delta t)$ at time $t+\Delta t$, in terms of the old, $f(\mathbf{x}, \mathbf{v}, t)$ at time t:

$$f(\mathbf{x}, \mathbf{v}, t+\Delta t) = \int_{\mathbf{x}'\mathbf{v}'} f(\mathbf{x}', \mathbf{v}', t)\, \mathbf{P}(\mathbf{x}, \mathbf{v}, \mathbf{x}', \mathbf{v}', \Delta t)\, \mathbf{dx}'\, \mathbf{dv}' \quad (1)$$

The "propagator" $\mathbf{P}(\mathbf{x}, \mathbf{v}, \mathbf{x}', \mathbf{v}', \Delta t)$ is proportional to the probability of a particle, which is at $(\mathbf{x}', \mathbf{v}')$ at time t, reaching $(\mathbf{x}, \mathbf{v})$ at time $t+\Delta t$. We now describe how the propagator is implemented numerically.

IIa. *Ballistic Motion*

In the absence of collisions the propagator can in principle be found by the method of characteristics. In practice, the way this is implemented is critical. Liouville's theorem states that in the absence of collisions, f is constant along a trajectory. The collisionless part of the propagator reflects this. There are several ways of understanding the basis for this propagator, but a good starting point is provided by the "water bag model" [6].

The water bag model is a representation of a distribution of particles in phase space which reflects the fact that the distribution function is incompressible (i.e., f is constant along a trajectory). Phase space is conceptually divided up into "water bags" such that within each bag, f is constant in phase space. The incompressibility property means that f is also constant in time, within the bag, although the bag moves through phase space as time progresses. The shape of the boundary of the bag changes as the bag moves, but constancy of f and conservation of particles mean that its volume is fixed.

The idea of the water bag is useful in understanding our numerical procedure. We divide phase space into a mesh of cells, as is usual in many numerical schemes. The simplest mesh will have uniform cell-width along each of its axes and will be constant in time. This mesh can be used to define the "water bags" at time t; each cell of the mesh corresponds to a "water bag" at this instant.

At time $t+\Delta t$, the water bags have moved. Suppose that we know the precise location of each bag at time $t+\Delta t$. Then the density in each bag is shared between several cells of the mesh, in the sense that the bag now overlaps several of the cells. We now need to redistribute the contents of the bag into these cells. The reallocation of that density into the cells of the mesh completes the time step, in the numerical scheme.

Calculating how much of the density in a given bag falls in each mesh cell at the end of the step is the essence of the ballistic (or collisionless) part of the move. This will now be described, using a simple approximation to the behavior of the bag. The one-dimensional version [4] is the basis for higher-dimensional calculations, so we begin by discussing this in detail. Higher dimensional cases, including velocity "dimensions" as well as space dimensions, are very similar.

In the convolution integral, Eq. (1), the "dummy" variable $\mathbf{x}'$ is in fact the "initial" position where the density is specified at time t. For this reason, the initial positions of the edges of the "water bag" in one dimension will be denoted $\mathbf{x}'_{\mathrm{L}}$ ($\mathbf{x}'_{\mathrm{H}}$) for the low (high) side, respectively. The final positions of the edges of the "bag" will similarly be denoted $\mathbf{x}_{\mathrm{L}}$ ($\mathbf{x}_{\mathrm{H}}$), and are found by integrating along trajectories starting at $\mathbf{x}'_{\mathrm{L}}$ ($\mathbf{x}'_{\mathrm{H}}$).

The initial positions of the edges coincides with the faces of a cell of the mesh, so if the initial cell is labeled by i',

$$\begin{aligned} \mathbf{x}'_{\mathrm{L}} &= (i'-1)\,\Delta\mathbf{x} \\ \mathbf{x}'_{\mathrm{H}} &= i'\,\Delta\mathbf{x}. \end{aligned} \quad (2)$$

The cells in which $\mathbf{x}_{\mathrm{L}}$ ($\mathbf{x}_{\mathrm{H}}$) lies will be labeled i_{L} (i_{H}), where on a uniform mesh

$$i_{\mathrm{L(H)}} = \mathrm{int}\left(\frac{\mathbf{x}_{\mathrm{L(H)}}}{\Delta x}\right) + 1. \quad (3)$$

If $i_{\mathrm{L}} > i_{\mathrm{H}}$, we reverse the labels L, H associated with the edges. The discussion below assumes that neither of the faces bounced or that they both bounced. If only one has bounced, we must find the point in the cell which just bounces in Δt and use it in the calculation of how we redistribute density.

$P(i, i')$ is the fraction of the density initially in cell i' which is placed in cell i. For a uniform cartesian mesh (i.e., a mesh with equal cell volumes) the contribution to the density in cell i due to initial cell i' is

$$\Delta f(i) = P(i, i')\, f(i'). \quad (4)$$

We now calculate $P(i, i')$ for the various cases which are possible. If $i_{\mathrm{L}} = i_{\mathrm{H}}$, then all the density from the initial cell i' is placed in the final cell $i = i_{\mathrm{L}} = i_{\mathrm{H}}$,

$$\begin{aligned} P(i, i') &= 1 \quad \text{for} \quad i = i_{\mathrm{L}} = i_{\mathrm{H}} \\ &= 0 \quad \text{for} \quad i \neq i_{\mathrm{L}} = i_{\mathrm{H}}. \end{aligned} \quad (5)$$

If $i_{\mathrm{H}} = i_{\mathrm{L}} + 1$, the "bag" is shared between two cells of the mesh. The numbers put in cells are determined by the overlap of the bag with the cells, as we shall see. The full length of the "bag" is $L_{\mathrm{b}} = x_{\mathrm{H}} - x_{\mathrm{L}}$ for all cases where none of the particles "bounce." The boundary between the cells whose indices are i_{L} and $i_{\mathrm{H}} = i_{\mathrm{L}} + 1$ is denoted $x_{i_{\mathrm{H}}} = (i_{\mathrm{H}} - 1)\,\Delta x = i_{\mathrm{L}}\,\Delta x$. The part of the bag within cell i_{L} extends from x_{L} to $x_{i_{\mathrm{H}}}$, so the fraction of the density in cell i_{L} is proportional to this length:

$$P(i, i') = (x_{i_{\mathrm{H}}} - x_{\mathrm{L}})/L_{\mathrm{b}}, \qquad i = i_{\mathrm{L}} = i_{\mathrm{H}} - 1. \quad (6a)$$

In cell i_H, the bag extends from x_{i_H} to x_H, so

$$P(i, i') = (x_H - x_{i_H})/L_b, \qquad i = i_H = i_L + 1 \tag{6b}$$

and $P(i, i') = 0$ otherwise.

If i_H exceeds i_L by more than one, then the bag extends across several cells at $t + \Delta t$. In cell i_L, the bag extends from x_L to x_{i_L+1}, so

$$P(i, i') = (x_{i_L+1} - x_L)/L_b, \qquad i = i_L. \tag{7a}$$

In cells between i_L and i_H, the bag extends the full cell width Δx, so that

$$P(i, i') = \Delta x/L_b, \qquad i_L < i < i_H. \tag{7b}$$

In cell i_H, the bag extends from x_{i_H} to x_H, so that

$$P(i, i') = (x_H - x_{i_H})/L_b, \qquad i = i_H. \tag{7c}$$

In more than one dimension, for example in two dimensions (x, y) with meshes labeled by integers (i, j) and uniform cell spacings $(\Delta x, \Delta y)$, the density placed in cell (i, j) coming from initial cell (i', j') is

$$\Delta f(i, j) = P(i, j, i', j')\, f(i', j'), \tag{8}$$

where

$$P(i, j, i', j') = P(i, i')\, P(j, j'). \tag{9}$$

This expression assumes only that the x-motion and the y-motion are separable. If not, it is straightforward to find a more accurate expression. Each coordinate and its corresponding velocity are not usually separable [5] but the motion in (x, v_x) can often be separated from that in (y, v_y). The total density at time $t + \Delta t$ is evaluated by creating an empty array and then adding the contributions $\Delta f(i, j)$ to it.

The simplest assumption which can be made in evaluating these fractions is that the entire bag moves at the same speed, so that $L_b = \Delta x$. Then $i_H = i_L + 1$, and according to Eq. (6),

$$P(i_H, i') = (x_H - x_{i_H})/\Delta x; \tag{9a}$$

$$P(i_L, i') = 1 - P(i_H, i'). \tag{9b}$$

These expressions were not accurate enough for the present work.

In all these cases, $\sum_i P(i, i') = 1$ for constant mesh spacing Δx. If Δx is not constant, conservation of particles implies that

$$(\Delta x_{i'})^{-1} \sum_i P(i, i')\, \Delta x_i = 1, \tag{10}$$

where Δx_i is the width of cell i and $\Delta x_{i'}$ is the width of the initial cell.

This discussion is clearly extendible to arbitrary numbers of variables. When velocity is included, the user has a choice whether to base the "overlap rule" on the overlap in velocity or the overlap in energy [5]. In this paper, we use two spatial coordinates and two components of velocity as independent variables, so the variables are (x, y, v_x, v_y) or (r, z, v_r, v_z).

It should also be noted that the cells can be moved in many ways. One can move the centers of cells, that is, use the electric field at the cell center and not allow cells to change their size or shape along any direction as they move. One other possibility (of many) is to move the centers of faces of the cells independently, using the local electric field and shape, but we usually force opposite faces to remain parallel to one another.

These approaches can be used in combination. For this work, in order to resolve the rapid variation in the electric field which takes place during sheath expansion, we move the spatial faces of each cell independently. However, we treat the cell as if the entire cell has the same velocity. We calculate the final locations of the spatial face centers (which start at a single velocity) accurately, in both space and velocity. For each spatial face of a cell, the entire face is moved in space in the same way as the center of the face. The extent of the face along the velocity axes (or energy axes, in a variant of the scheme [5]) is treated as constant, so if the face center accelerates, the entire face accelerates accordingly.

If this is done, a relatively coarse spatial mesh is adequate. The steady state, on the other hand, has more gentle variations in the electric field, so the front and back faces can be given an identical "move" with the same final velocity. Indeed, this or some other measure such as damping the electric field [5] can be helpful to suppress certain numerical oscillations which occur when long time steps are used.

IIb. *Collisions*

Collisions will be considered next. The collision process can be handled in one spatial cell at a time, since collisions do not move directly particles from one spatial location to another. As a result, the treatment of collisions does not depend on the number of spatial coordinates employed. Two basic approaches have been proposed in the past [4, 5]. The simplest version of the collision operation is briefly reviewed here.

The collision process in this formulation is included in a conceptually separate stage from the "ballistic" process described above. It may be performed at the same time, that is, in the same pass over the array of phase space cells, in some cases [5]. For the sake of this discussion, we assume that the "ballistic move" is done "first," however.

After the "ballistic" move, we know the distribution in each spatial cell; omitting the spatial dependence we write this as $f(\mathbf{v})$. The collision frequency is $\nu(\mathbf{v})$, so the density

scattered out of the cell in time Δt is $f(\mathbf{v})\,\upsilon(\mathbf{v})\,\Delta t$. We limit Δt so that $\upsilon(\mathbf{v})\,\Delta t \ll 1$. In all of this discussion of collisions, $\mathbf{v}$ is used to indicate the discrete values of $\mathbf{v}$ represented on the velocity mesh.

We normally remove that fraction of the density which is scattered from all velocities in this spatial location, before replacing any (but see Ref. [5]). This density must then be redistributed on the mesh. When all the scattered density which will leave cells in this spatial location has been removed, it is stored according to the final velocity it will have; in the case of isotropic scattering it can be stored according to speed. In the case considered here, where ions are scattered as a Maxwellian, it is only necessary to store the total density scattered in that spatial cell. If $\Delta f(\mathbf{v})$ is the total number to be replaced at $\mathbf{v}$, then this is added to the $f(\mathbf{v})$ obtained after all the scattered particles are removed.

For the specific case of a Maxwellian distribution of scattered particles, we first construct a Maxwellian distribution $F_{\max}(\mathbf{v})$ at the required temperature T, normalized to a density of one on the numerical mesh. If the particles being replaced contribute a density d_r, then we add density $d_r F_{\max}(\mathbf{v})$ to each cell.

III. MULTI-DIMENSIONAL ELECTROSTATICS

One of the novel features of this work is the use of more than one spatial dimension in an electrostatically self-consistent solution of Boltzmann's equation. In order to allow the simulation to proceed on the ion time scale, we assume the electrons obey the Boltzmann relation,

$$n_e = n_0 \exp\left(\frac{q\Phi}{kT_e}\right), \tag{11}$$

where T_e is the electron temperature. This is used to solve Poisson's equation,

$$\nabla^2\Phi = \frac{e}{\varepsilon_0}\,[n_e - n_i]. \tag{12}$$

In two or more dimensions, the equations must be solved by an iterative technique, and it is highly desirable to use sparse matrix techniques given a mesh of the scale envisaged here. This has been done, using a flexible numerical package described elsewhere [7].

IV. RESULTS AND DISCUSSION

We now consider the results of the simulation. Calculating the kinetics is much faster than solving Poisson's equation. One time step of the CS on a (20, 20, 30, 30) mesh in (r, z, v_r, v_z) takes about two seconds on an HP 735 workstation. In this case the step $\Delta t = 10^{-7}$, and with $\Delta v = 3 \times 10^6$ m/s, the fastest ions travel about 5 cm in Δt. The

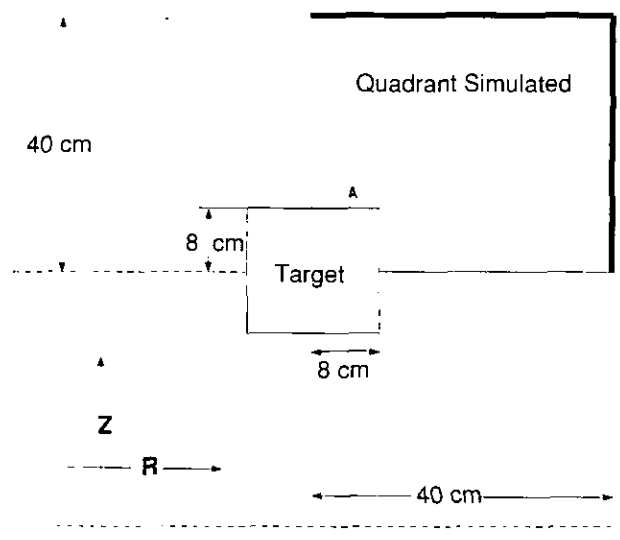

FIG. 1. The geometry of the system.

Poisson equation, Eq. (12), is solved by a Newton method, which needs 3–6 iterations at each time step to converge to very high accuracy, each iteration taking 5–20 s.

The geometry of the system used to illustrate the problem is shown in Fig. 1, showing the target and the vessel containing the plasma. T_e is taken to be 0.57 eV. The peak initial ion density is $n_i = 7 \times 10^9\ \mathrm{cm}^{-3}$. The coordinate system is cylindrical, so the target can be thought of as the cross section of a cylinder.

The parameters were chosen for the simulation for their consistency with typical PSII experiments [8, 9]. The gas is assumed to be argon. Ion-neutral charge exchange collisions were included, assuming a mean free path of 6 cm. The target was ramped down from being essentially grounded to a voltage of -30 kV. The ramping rate used is 3 kV/μs, which is reasonable for PSII. However, a detailed comparison with experiment is not presented here.

Dirichlet boundary conditions were used for the potential on both the target and the outer walls; that is, Φ was fixed there. Since the electrostatic calculation used the Boltzmann assumption for the electron density, the peak voltage could not go above zero for that would allow an electron density greater than the imposed peak density. To ensure that the peak voltage was below zero, but large enough that the electron density was close to the peak ion density, the outer wall and the target prior to ramping were both held at -1 V, since the plasma potential relative to a grounded wall is typically a few times the electron temperature.

As the electrostatic potential applied to the target is ramped down to very low values, the sheath expands rapidly due to both the increase in potential and the decrease in ion density as the ions move rapidly towards the

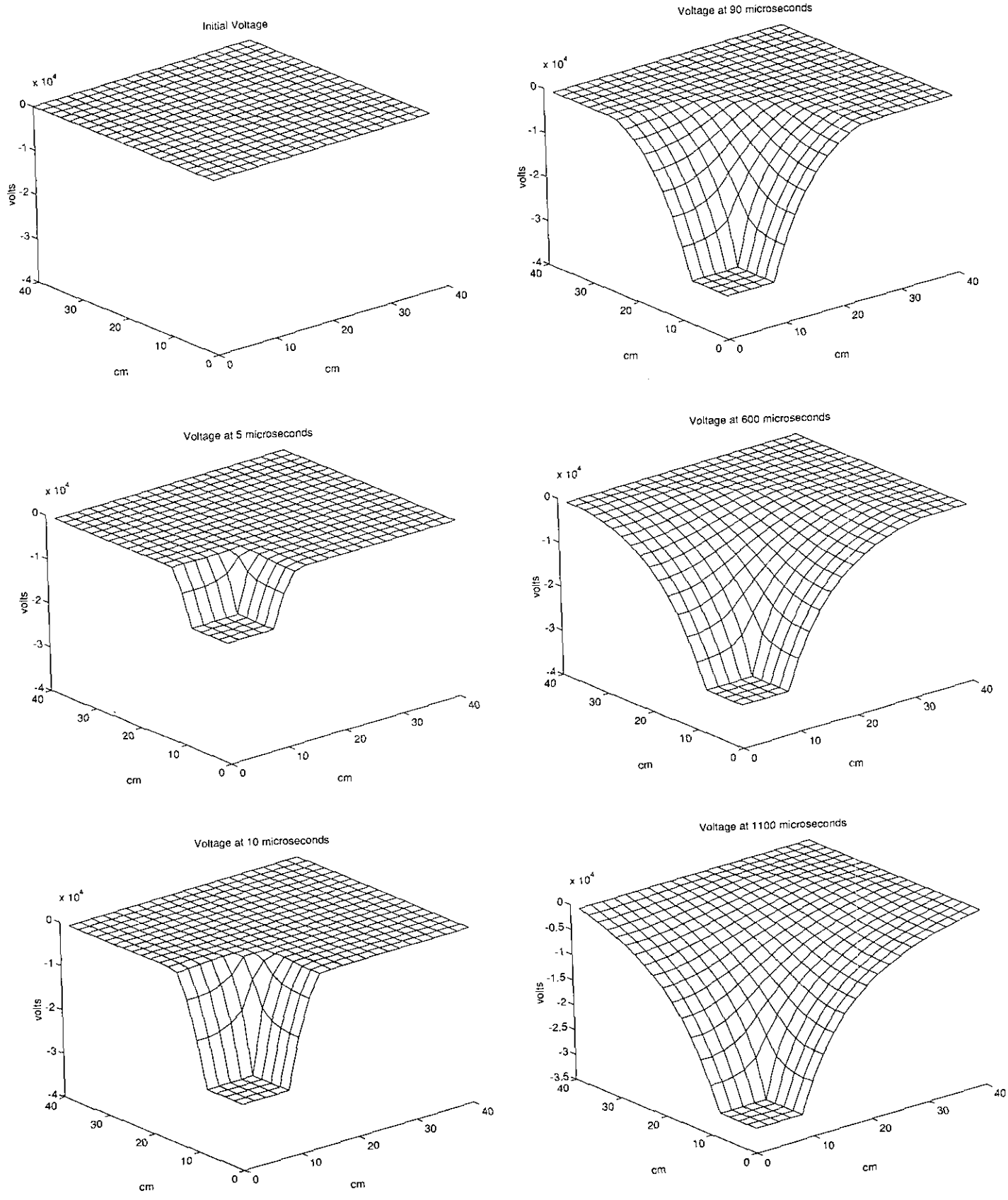

FIG. 2. The electrostatic potential at various times.

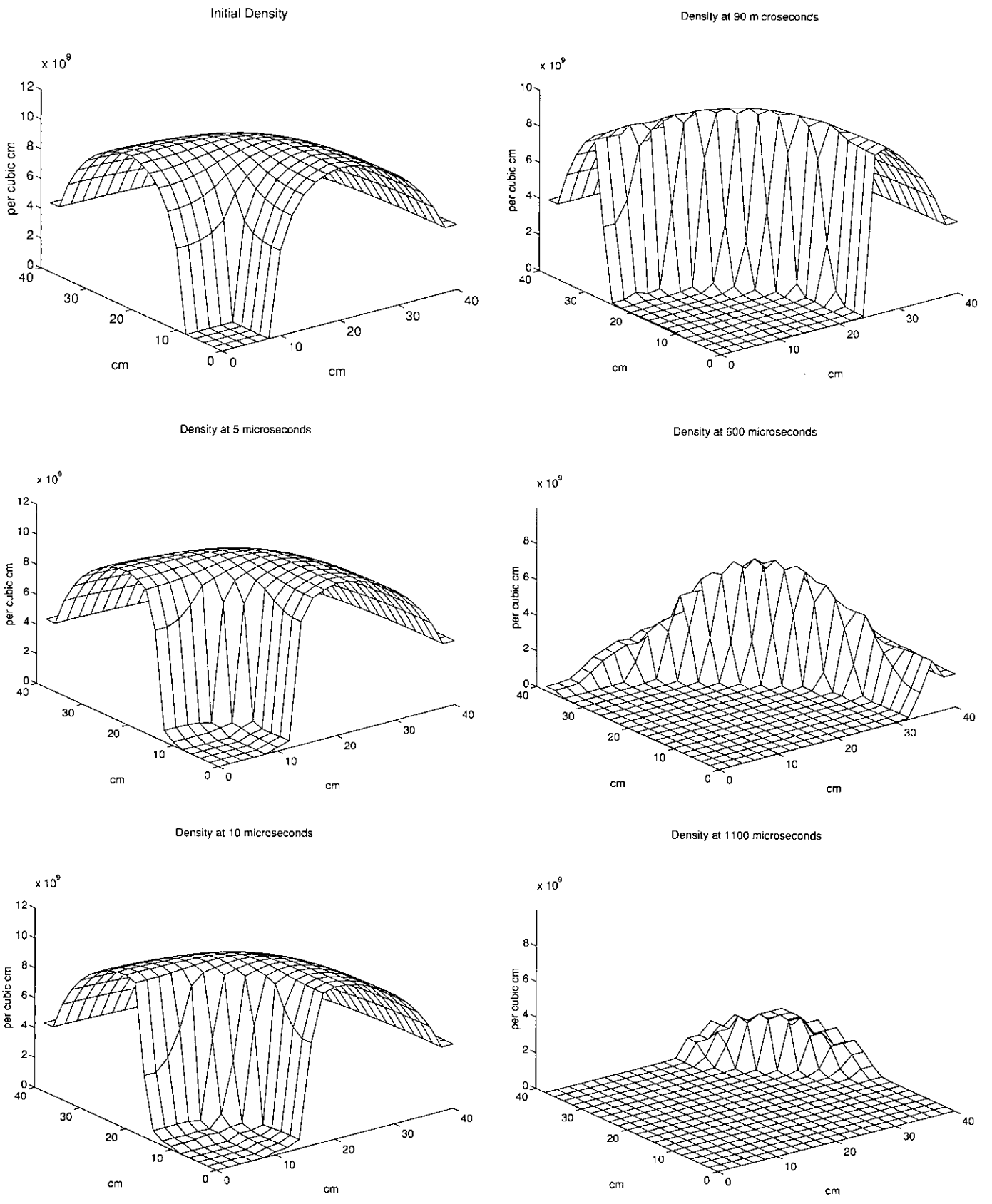

FIG. 3. The ion density at various times.

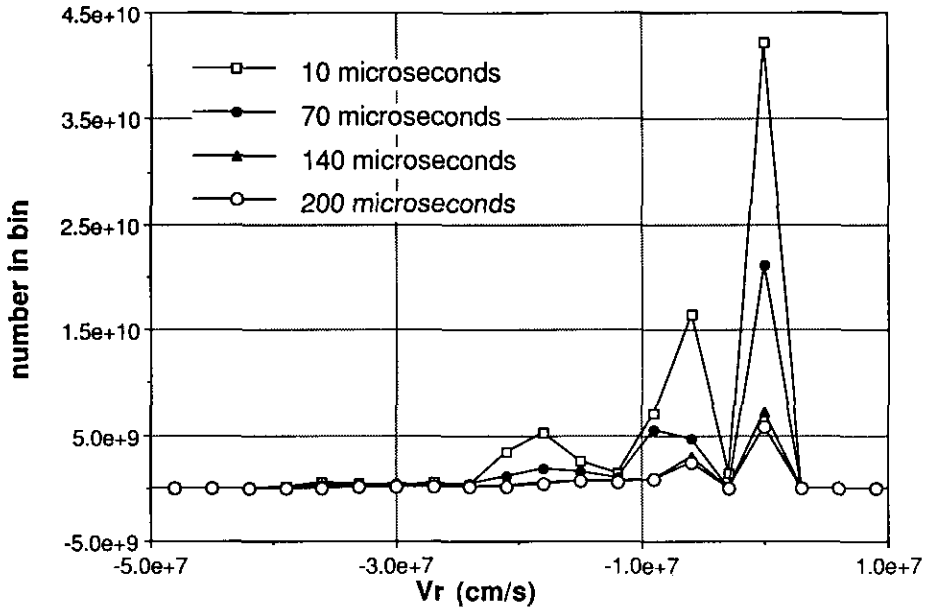

FIG. 4. The ion velocity distribution with respect to Vr at a point in the discharge, at various times. The location of the point corresponds to point A in Fig. 1. This has been integrated over all Vz.

target. The ions accelerate so rapidly once they are in the sheath that there is a dramatic density change on crossing the sheath edge. The density in the sheath is very small compared to the initial value, whereas the density outside the sheath is hardly changed. (In the outer sheath there is a noticeable drop in ion density, also.)

The ion density shown does indeed rapidly develop a profound hollow as the sheath moves through it. The density and electrostatic potential are shown versus time in Figs. 2 and 3. The density of the corner of the mesh farthest from the target is the last to be removed, simply because it is the most distant. The main quantity of interest is the velocity distribution function f_i shown in Figs. 4 and 5 for point A in Fig. 1. The functions displayed are for various times that occur after the sheath edge has passed point A. The ions at this point can be subdivided into two distinct groups, those that have recently undergone a charge exchange collision and those that have not. Those that have

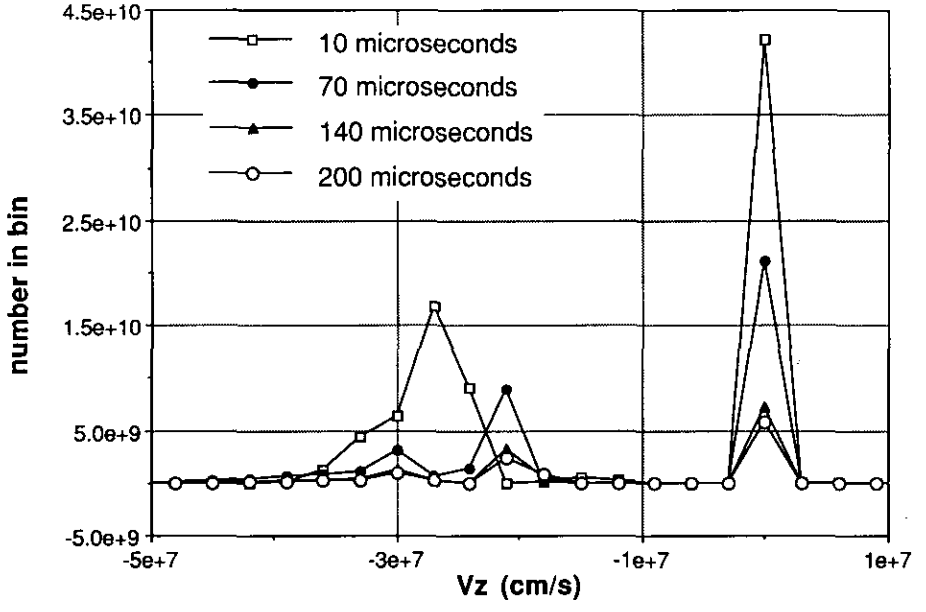

FIG. 5. The ion velocity distribution with respect to Vz at a point in the discharge, at various times. The location of the point corresponds to point A in Fig. 1. This has been integrated over all Vr.

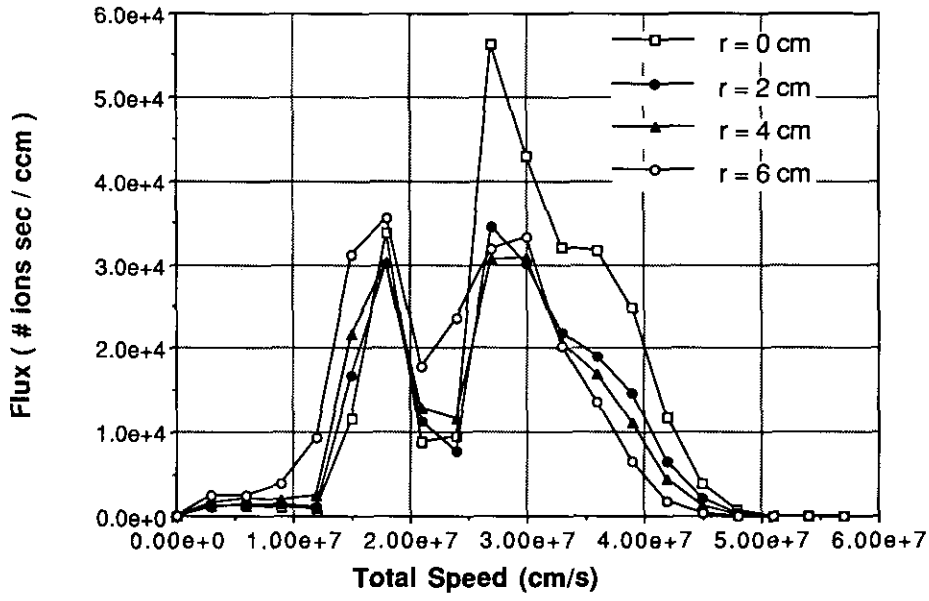

FIG. 6. The flux per unit velocity of ions striking the target, at different radii along the target's top surface.

undergone collisions have been redistributed with a Maxwellian distribution about the zero velocity cell and therefore account for the peak at zero. Those that have not had a collision recently are spread out over the non-zero velocity cells. The nonzero energy ions are not quite a "beam," since ions originate from points with different potential energies. The distribution corresponding to the earliest time is more sharply peaked than those from later times, and this is precisely because the ions have a smaller range of potential energies from which they could have originated. Knowledge of f_i enables the time-integrated flux striking different points on the target to be obtained as a function of ion speed (Fig. 6).

As is evident in Figs. 4 and 5, there are significant components of velocity in both r and z. Point A is nearest a surface which is parallel to the r direction, so one would expect the velocity in the z direction to be large. The fact that the tail of a transverse velocity distribution is of a similar magnitude to that of the velocity normal to the surface is of interest, for this indicates that a large number of the incident ions will have trajectories that are far from perpendicular to the surface of the target. It should be noted that the geometry of this target is relatively uncomplicated, and the fact that this large a deviation from normal trajectories should occur is indicative that effects of a similar nature are likely for more realistic targets.

The time-integrated flux per unit velocity (Fig. 6) varies noticeably across the target. The two peaks correspond to ions that have recently undergone a collision and ions that have not, as was the case for the velocity distributions discussed above. The peaks are much broader than those of the velocity plots because we have combined both components of the velocity to obtain the total ion speed, which is the horizontal axis in Fig. 6. As expected, total flux is principally concentrated at high energies for a mean free path of 6 cm, but the peak is very broad. From this profile, the depth profile of implanted ions can readily be obtained.

In summary, we have demonstrated an electrostatically self-consistent solution of the ion kinetic equation, in two spatial dimensions. The method, which is very efficient and easy to implement, was applied to a problem in "plasma source ion implantation." The distribution function of ions and the electrostatic potential were found, as well as the velocity spectrum of implanted ions for an applied target voltage which was rapidly ramped down to a very large negative value.

ACKNOWLEDGMENT

This work was supported in part by NSF under Contract ECD-8721545.

REFERENCES

1. R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (McGraw–Hill, New York, 1981).
2. J. W. Eastwood, *Comput. Phys. Commun.* **44**, 73 (1987).
3. R. Boswell, private communication, 1992; V. Vahedi, M. A. Lieberman, M. V. Alves, J. P. Verboncoeur, and C. K. Birdsall, *J. Appl. Phys.* **69**, 2008 (1991).
4. W. N. G. Hitchon, D. J. Koch, and J. B. Adams, *J. Comput. Phys.* **83**, 79 (1989).
5. W. N. G. Hitchon, G. J. Parker, and J. E. Lawler, *IEEE Trans. Plasma Sci.* **21**, 228 (1993).
6. H. L. Berk and K. V. Roberts, *Methods Comput. Phys.* **9**, 88 (1970).
7. K. M. Kramer and W. N. G. Hitchon, A highly flexible tool for large-scale computational problems in applied physics, to be published.
8. J. R. Conrad, J. L. Radke, R. A. Dodd, F. J. Worzala, and N. C. Tran, *J. Appl. Phys.* **62**, 4591 (1987).
9. J. R. Conrad, R. A. Dodd, F. J. Worzala, and X. Qui, *Surf. Coatings Technical.* **36**, 927 (1988).