Computing the Paschen curve for argon with speed-limited particle-in-cell simulation

Joseph G. Theis,[1, a)] Gregory R. Werner,[1] Thomas G. Jenkins,[2] and John R. Cary[1, 2]

[1)] *Center for Integrated Plasma Studies, University of Colorado, Boulder, Colorado 80309, USA*

[2)] *Tech-X Corporation, 5621 Arapahoe Avenue Suite A, Boulder, Colorado 80303, USA*

(Dated: 2 July 2021)

Upon inclusion of collisions, the speed-limited particle-in-cell (SLPIC) simulation method successfully computed the Paschen curve for argon. The 1D3V simulations modelled an electron cascade across an argon-filled capacitor, including electron-neutral ionization, electron-neutral elastic collisions, electron-neutral excitation, and ion-induced secondary electron emission. In electrical breakdown, the timescale difference between ion and electron motion makes traditional particle-in-cell (PIC) methods computationally slow. To decrease this timescale difference and speed up computation, we used SLPIC, a time-domain algorithm that limits the speed of the fastest electrons in the simulation. The SLPIC algorithm facilitates a straightforward, fully-kinetic treatment of dynamics and collisions. SLPIC was as accurate as PIC, but ran up to 200 times faster. SLPIC accurately computed the Paschen curve for argon over three orders of magnitude in pressure.

[a)]joseph.theis@colorado.edu

I. INTRODUCTION

Townsend discharge is the conduction of electric current in a gas due to a cascade of ionizing collisions.[1] In typical discharge experiments, the gas is enclosed between the electrodes of a capacitor, and there is an initial source of free electrons. The electrons accelerate in an applied electric field and ionize the neutral atoms, creating more electrons, which ionize more atoms. The ions accelerate towards and impact the cathode, where they can produce secondary electrons. When the combined effects of ionizing collisions and ion-induced cathode emission yield more electrons at the cathode than initially seeded, a self-sustaining discharge is triggered. The voltage difference at which this occurs is the breakdown voltage.

The breakdown voltage V_b of a gas-filled capacitor is (to a good approximation) a function of pd, where p is the pressure of the gas and d is the distance between the electrodes. A Paschen curve is the relationship between V_b and pd for a given gas.[2] Paschen curves have a minimum V_b at some intermediate pd and higher V_b at the extremes (this shape is shown later in Fig. 2). This characteristic shape is due to the competition between ionizing collisions, which multiply the number of free electrons, and non-ionizing collisions, which prevent electrons from gaining enough energy to ionize atoms. At low pd, the ionization mean free path is too long for a sufficient number of ionizations to occur during crossing. At high pd, the non-ionizing mean free path is too short for electrons to reach the ionization energy. Paschen curves will be developed further in §II A.

Electrical discharge and breakdown have been extensively studied both for a multitude of applications, including lighting,[3] plasma processing,[4] and lasers,[5] as well as to avoid detrimental effects, including damage to electrical devices,[6] explosions in chemical processing facilities,[7] and damage to aerospace systems.[8] Following Townsend's description of the mechanism behind gas discharge,[1] various models and simulation methods have been employed to study the phenomena.[9] Analytical models have proven effective for computing breakdown and steady-state current amplification in simple geometries;[9] however, understanding the particle dynamics of discharge and handling complex geometries requires simulation.

Particle-in-cell (PIC) and fluid methods have been the primary simulation techniques applied to model gas discharges.[4,10] Fluid approaches evolve the mean properties of a particle species and can therefore handle situations involving many particles and mean free paths that are small compared with the system size. However, the fluid approaches assume Maxwellian

particle velocity distributions, while discharges often have non-Maxwellian distributions.[10]

PIC methods evolve the distribution function $f(x, v, t)$ in time via the method of characteristics and, therefore, can simulate arbitrary distribution functions.[11] Collisions can be included in PIC using the PIC Monte-Carlo Collisions framework (PIC-MCC).[11–13] PIC-MCC will be discussed further in §III. PIC-MCC requires that the smallest mean free path be resolved by the grid cell size Δx. Further, in PIC, the timestep Δt is limited according to $\Delta t \lesssim \Delta x / v_{\max}$, where v_{max} is the maximum particle speed. In gas discharge simulations, the large mass difference between the ions and the electrons results in ion timescales much greater than the timestep imposed by the maximum electron speed. This makes PIC-MCC simulations computationally slow. Because PIC-MCC is slow, variations of PIC simulation have been applied to discharge. One approach is treating the ions as immobile to eliminate the timescale mismatch,[14,15] but this technique fails when ion motion becomes relevant as is the case for Townsend discharge. Another technique is electron sub-cycling, in which the timestep for the electron is smaller than that of the ion,[16] but this technique yields at most a factor of two speed-up. Hybrid methods combine PIC and fluid approaches (e.g., with fluid electrons but PIC ions),[10] but they fail to capture the full velocity distribution as PIC does.

In this work, we apply the speed-limited particle-in-cell (SLPIC) simulation method[17] to electrical breakdown in argon. SLPIC is a time-domain algorithm that limits the speed of the fastest particles in the simulation to enable larger time steps and, therefore, faster computing times. The SLPIC algorithm facilitates a straightforward, fully-kinetic treatment of effects such as secondary emission at the cathode and collisions. A detailed introduction to SLPIC is provided in §II B. This work is the first demonstration of the integration of SLPIC particles with PIC-MCC, showing that speed-limiting techniques can be used in collisional low-temperature plasma discharges. We chose to simulate electrical breakdown, and, specifically, Paschen's law, because multiple collision mechanisms must be correctly simulated to compute the breakdown voltage accurately. We were also able to quantify the speed-up in runtime of the SLPIC simulations relative to the regular PIC simulations.

In this paper, we begin with a review of Paschen curves and SLPIC. We then introduce the SLPIC Monte Carlo Collisions framework in §III. The simulation design and parameters will be provided along with our methodology for determining breakdown in §IV. In §V, we compare our simulation results (a Paschen curve) with experimental data and quantify the

computational speed-up of the SLPIC algorithm relative to normal PIC.

II. BACKGROUND

A. Paschen Curve

Electrical breakdown in a gas-filled capacitor is caused by the ionization of the neutral gas by electrons and the secondary emission of electrons at the cathode due to ion impact. The average number of ions produced per electron per unit length travelled is termed[1] Townsend's first ionization coefficient α. The average number of secondary electrons emitted per ion impact is Townsend's second ionization coefficient γ_{se}. The coefficient α depends on the voltage V across the gap and the collision cross-sections. The coefficient γ_{se} depends on the ion species and the electrode material and treatment. When $\gamma_{se}(e^{\alpha d} - 1) > 1$ breakdown occurs.

Paschen curves relate the breakdown voltage V_b of a gas-filled capacitor to the product of the gas pressure p and the distance d between the electrodes.[2] Paschen curves are not monotonic due to a competition between electrons gaining energy in the electric field and losing energy in collisions. Paschen curves have a characteristic minimum at an intermediate pd. At low pd, most electrons stream through the capacitor without ionizing enough atoms. At high pd, few electrons reach the threshold ionization energy due to frequent non-ionizing collisions. Paschen's law is a function relating V_b to pd:

$$V_b = \frac{Bpd}{\ln(Apd) - \ln[\ln(1 + 1/\gamma_{se})]}. \tag{1}$$

Paschen's law assumes that, except for the initial seed electrons, ion impact at the cathode and electron-neutral ionizations are the only sources of electrons, and that α can be expressed as:

$$\frac{\alpha}{n_g} = kTA \exp\left(\frac{-kTBn_g}{E}\right) \tag{2}$$

where kT is the thermal energy, n_g is the neutral gas density, and $E = V/d$ is the constant electric field. A and B are related to the collision cross-sections and are determined by fitting Eq. (2) (which gives the field-intensified ionization cross-section α/n_g as a function of the reduced field E/n_g) to the experimental results for a given gas. In Paschen's law, A and B are assumed to be constant for a given gas species. Paschen's law will be plotted in Fig. 2, with the values of A, B, and γ_{se} taken from Lieberman and Lichtenberg[4].

Different gases yield different Paschen curves due to different cross-sections, ionization energies, and secondary emission yields at the cathode. In this work, we simulated a neutral argon gas. Argon has been studied extensively in the context of electrical breakdown, including simulation,[10,18,19] and experimental work.[9,20,21] Research has been focused on identifying the mechanisms responsible for breakdown and the regimes in which these mechanisms are relevant. In this work, we will compare our simulation results to experimental results compiled by Phelps and Petrovic[9].

B. SLPIC

Before describing how collisions can be implemented within the SLPIC method in §III, we review the derivation of SLPIC, following Werner *et al.*[17]. PIC numerically evolves a particle distribution $f(\mathbf{x}, \mathbf{v}, t)$ according to the Vlasov equation:

$$\partial_t f(\mathbf{x}, \mathbf{v}, t) = -\mathbf{v} \cdot \partial_{\mathbf{x}} f(\mathbf{x}, \mathbf{v}, t) - \mathbf{a} \cdot \partial_{\mathbf{v}} f(\mathbf{x}, \mathbf{v}, t), \tag{3}$$

where the acceleration $\mathbf{a} = \mathbf{a}(\mathbf{x}, \mathbf{v}, t)$ is typically determined by electromagnetic fields. SLPIC numerically evolves $f(\mathbf{x}, \mathbf{v}, t)$ according to an *approximate* Vlasov equation. We multiply Eq. (3) by a yet-to-be-chosen function $\beta(v)$ (where $v = \|\mathbf{v}\|$), which will turn out to limit particle speeds,

$$\beta \partial_t f(\mathbf{x}, \mathbf{v}, t) = -\beta \mathbf{v} \cdot \partial_{\mathbf{x}} f(\mathbf{x}, \mathbf{v}, t) - \beta \mathbf{a} \cdot \partial_{\mathbf{v}} f(\mathbf{x}, \mathbf{v}, t), \tag{4}$$

and then make the critical "speed-limiting" approximation:

$$\beta \partial_t f \approx \partial_t f. \tag{5}$$

This approximation is valid if either (1) $\partial_t f(\mathbf{x}, \mathbf{v}, t) \approx 0$ or (2) $\beta(v) \approx 1$.[22] Thus we arrive at the "speed-limited" approximation to the Vlasov equation

$$\partial_t f(\mathbf{x}, \mathbf{v}, t) = -\beta \mathbf{v} \cdot \partial_{\mathbf{x}} f(\mathbf{x}, \mathbf{v}, t) - \beta \mathbf{a} \cdot \partial_{\mathbf{v}} f(\mathbf{x}, \mathbf{v}, t). \tag{6}$$

In the special case of a steady state (i.e., $\partial_t f \equiv 0$), the speed-limiting approximation is exact and SLPIC, though different from PIC, is as accurate as PIC. For the special choice of $\beta(v) \equiv 1$, SLPIC is exactly the same as PIC.

Just as PIC simulation evolves the Vlasov equation via the method of characteristics, SLPIC evolves Eq. (6). We posit a solution that is a sum over macroparticles p that follow

trajectories $[\mathbf{x}_p(t), \mathbf{v}_p(t)]$ with weights $w_p(t)$ (a macro-electron with weight w_p represents w_p electrons; i.e., it has mass $w_p m_e$ and charge $-w_p e$):

$$f(\mathbf{x}, \mathbf{v}, t) = \sum_p w_p(t) S[\mathbf{x} - \mathbf{x}_p(t)] \delta^3[\mathbf{v} - \mathbf{v}_p(t)] \tag{7}$$

where S is the particle shape function[11] and δ the Dirac delta function. Unlike PIC, SLPIC requires the particle weight to vary in time. Plugging Eq. (7) into Eq. (6), we find that it yields a solution if:

$$\dot{\mathbf{x}}_p = \beta(v_p)\mathbf{v}_p \tag{8}$$

$$\dot{\mathbf{v}}_p = \beta(v_p)\mathbf{a}(\mathbf{x}_p, \mathbf{v}_p, t) \tag{9}$$

$$\dot{w}_p = w_p[\partial_{\mathbf{x}} \cdot (\beta\mathbf{v}) + \partial_{\mathbf{v}} \cdot (\beta\mathbf{a})] = \beta w_p[\partial_{\mathbf{x}} \cdot \mathbf{v} + \partial_{\mathbf{v}} \cdot \mathbf{a}] + w_p \mathbf{a} \cdot \partial_{\mathbf{v}}\beta = w_p\dot{\beta}/\beta \tag{10}$$

where we have assumed that $\partial_{\mathbf{x}} \cdot \mathbf{v} + \partial_{\mathbf{v}} \cdot \mathbf{a} = 0$—i.e., the true motion must be phase-space-volume preserving (Liouville's theorem proves this for any Hamiltonian motion). We have also substituted $\dot{\beta} \equiv (d/dt)\beta(v_p(t)) = \beta\mathbf{a} \cdot \partial_{\mathbf{v}}\beta$.

When $\beta(v_p) = 1$, these are the usual (PIC) equations of motion. When $0 < \beta(v_p) < 1$, a SLPIC macroparticle, representing physical particles with true velocity $\mathbf{v}_p$, travels with pseudo-velocity $\dot{\mathbf{x}}_p = \beta(v_p)\mathbf{v}_p$ and pseudo-acceleration $\dot{\mathbf{v}}_p = \beta(v_p)\mathbf{a}(\mathbf{x}_p(t), \mathbf{v}_p(t), t)$. The SLPIC macroparticle follows the same path through phase space as the physical particles it represents, but slower; e.g., in a short time Δt, the macroparticle moves $(\mathbf{v}_p, \mathbf{a})\beta\Delta t$, while a physical particle moves $(\mathbf{v}_p, \mathbf{a})\Delta t$. For example, a SLPIC electron in a uniform magnetic field will have the correct gyroradius, but its gyrofrequency will be (unphysically) reduced by a factor β_{v_p}.

Choosing $\beta(v_p) < 1$ invokes the SLPIC approximation [Eq. (5)], reducing accuracy; however, it limits the simulated speed of macroparticles from v_p to $\dot{x} = \beta(v_p)v_p$, which allows larger timesteps, hence faster simulation. In most cases, the PIC timestep must be smaller than $\Delta t \lesssim \Delta x/v_{\max}$, where Δx is the grid cell size and $v_{\max}$ is the maximum particle speed. With SLPIC, a larger timestep is allowed: $\Delta t \lesssim \Delta x/[\beta(v_{\max})v_{\max}]$. Typically, we choose a speed limit $v_0 \ll v_{\max}$, and then define $\beta(v_p) \equiv v_0/v_p$ to ensure that for all particles, $\|\dot{\mathbf{x}}\| \leq v_0$, so that we can use $\Delta t \approx \Delta x/v_0 \gg \Delta x/v_{\max}$. For particles with $v_p < v_0$, however, we define $\beta(v_p) \equiv 1$ so that they experience "true" motion. In this way, we speed-limit only the particles (with $v_p > v_0$) that need to be speed-limited to allow a large timestep, $\Delta t = v_0/\Delta x$. It turns out that speed-limiting also reduces the plasma frequency so that it is

resolved by the large Δt and poses no instability.[17,23] For simplicity, we will use this specific $\beta(v_p)$ for the rest of the paper: $\beta(v_p) = v_0/v_p$ for $v_p \geq v_0$, and $\beta(v_p) = 1$ for $v_p < v_0$, or, equivalently:

$$\beta(v_p) \equiv \Theta(v_p - v_0)v_0/v_p + \Theta(v_0 - v_p) \tag{11}$$

where Θ is the Heaviside step function (and $v_p = \|\mathbf{v}\|$).

As a macroparticle accelerates so that $v_p > v_0$, its simulated speed $\dot{x}_p = \beta v_p$ remains at v_0, and its weight $w_p(t)$ decreases. The decrease in $w_p(t)$ reflects the fact that the physical particle density represented by the macroparticle at time t decreases, and the macroparticle represents fewer physical particles at time $t + \Delta t$. In this way, SLPIC can correctly simulate a steady-state beam within which particles accelerate such that their density decreases (like cars exiting a steady-state traffic jam). In SLPIC, a beam macroparticle with $v_p > v_0$ experiences an acceleration and its true velocity v_p increases, but its pseudo-velocity $\dot{x}_p = \beta v_p = v_0$ does not, so the density of macroparticles throughout the beam remains uniform. The decrease in physical density thus results from the decrease in macroparticle weight, and not (as in PIC) from the decrease in macroparticle density. While SLPIC may sometimes result in a more uniform macroparticle density, this is an unintended bonus and it may not always be the case. SLPIC is orthogonal to techniques for managing macroparticles weights/numbers for efficiency and accuracy.[24,25]

Therefore, as v_p increases above v_0, $\beta(v_p)$ decreases, and so too must w_p decrease. Equation (10) shows that $w_p/\beta(v_p)$ is constant over any particle trajectory; we define this constant to be $w_{j,p} \equiv w_p/\beta(v_p)$. Here we choose the subscript j because $w_{j,p}$ facilitates calculation of the flux density distribution $\mathbf{j}(\mathbf{x}, \mathbf{v}, t)$.

Simulating macroparticles following Eqs. (8–10), we can compute the phase-space density distribution function f at any time via Eq. (7). However, with SLPIC it is often especially useful to consider particle fluxes. We can compute the flux density distribution, $\mathbf{j}(\mathbf{x}, \mathbf{v}, t) = \mathbf{v} f(\mathbf{x}, \mathbf{v}, t)$, as follows:

$$\mathbf{j}(\mathbf{x}, \mathbf{v}, t) = \sum_p \mathbf{v}_p w_p(t) S[\mathbf{x} - \mathbf{x}_p(t)]\delta^3[\mathbf{v} - \mathbf{v}_p(t)] = \sum_p \beta(v_p)\mathbf{v}_p w_{j,p} S[\mathbf{x} - \mathbf{x}_p(t)]\delta^3[\mathbf{v} - \mathbf{v}_p(t)]. \tag{12}$$

To estimate the flux $\mathbf{j} \cdot \hat{\mathbf{n}} dA$ through some surface element $\hat{\mathbf{n}} dA$ from the macroparticles crossing it in time Δt, we consider that a macroparticle with $\dot{\mathbf{x}}$ will cross the surface if it

is within a distance $\dot{\mathbf{x}}\Delta t \cdot \hat{\mathbf{n}}$ of the surface. From Eq. (12) we see that, since $\dot{\mathbf{x}} = \beta(v_p)\mathbf{v}_p$, the flux (integrated over time Δt) through the surface element is simply the sum of the $w_{j,p}$ over all macroparticles crossing the surface in Δt.

Because the $w_{j,p}$ are constant in time, a macroparticle always represents the same physical flux whenever it crosses a surface. If a macroparticle p has $w_{j,p} = 100$, then every time it crosses a surface, it represents 100 physical particles crossing the surface. This statement is trivial in PIC, but in SLPIC, where $w_p(t)$ varies in time, it is nontrivial and very useful. For example, it ensures that, if a macroparticle enters into some volume V and later exits that volume, the time-integrated flux through the surface (due to that particle) is zero.

At this point readers might suspect a paradox in the SLPIC treatment arising from a difference between w_p and $w_{j,p}$. Consider a macroparticle with $v_p = 10v_0$, $\beta(v_p) = 0.1$, $w_{j,p} = 100$, and $w_p = 10$, and suppose it crosses a surface to enter volume V during time interval Δt. That macroparticle represents $w_p = 10$ physical particles in volume V; however, it represented $w_{j,p} = 100$ physical particles crossing the surface into V. This counterintuitive behavior is correct for SLPIC. It is counterintuitive because PIC has led us to assume that a macroparticle with w_p and $(\mathbf{x}_p, \mathbf{v}_p)$ represents a swarm of w_p physical particles within the volume $d^3x\, d^3v$ around $(\mathbf{x}_p, \mathbf{v}_p)$, and that those physical particles travel roughly with the macroparticle to $(\mathbf{x}_p + \dot{\mathbf{x}}_p\Delta t, \mathbf{v}_p + \dot{\mathbf{v}}_p\Delta t)$ over time Δt. Fundamentally, however, a macroparticle represents a chunk of the distribution $f d^3x\, d^3v$, and SLPIC decouples macroparticles from the physical particles they represent. A speed-limited SLPIC macroparticle moves in phase space with $\dot{\mathbf{x}}_p$ and $\dot{\mathbf{v}}_p$ slower than the physical particles it represents. Thus the same macroparticle may represent one set of physical particles at time t and a different set of particles at time $t + \Delta t$.

In the above example, the density represented by the SLPIC macroparticle corresponds to $w_p = 10$ physical particles that are near $[\mathbf{x}_p(t), \mathbf{v}_p(t)]$ at time t; however, over time interval Δt, the physical flux represented by the macroparticle includes all the physical particles that would would be near $[\mathbf{x}_p(t'), \mathbf{v}_p(t')]$ at any time $t' \in [t, t + \Delta t]$. Since the physical particles travel 10 times faster than the macroparticle, the physical flux is $10w_p(t)/\Delta t = w_{j,p}/\Delta t$.

It is important to remember that SLPIC is accurate only if $f(\mathbf{x}, \mathbf{v}, t)$ changes sufficiently slowly—and it is as accurate as PIC in the steady-state limit. The $f(\mathbf{x}, \mathbf{v}, t)$ describing a single particle with $v > v_0$ is not a slowly-changing function; therefore, one typically cannot verify SLPIC based on single-particle thought experiments. In the above single-particle

example, the continuity equation is violated because $f(\mathbf{x}, \mathbf{v}, t)$ changes too rapidly for the SLPIC approximation to be valid.

Exactly what constitutes "sufficiently slowly" for SLPIC accuracy remains an open question. We have previously observed that for (Langmuir) wave-particle interaction in 1D, "sufficiently slowly" means that the wave phase velocity must be slower than the SLPIC speed limit for accurate simulation[17,23]. Similarly, although studying SLPIC in magnetized plasmas is beyond the scope of this work, we imagine that the electron-cyclotron resonance interaction will be too fast for accurate simulation, as long as SLPIC reduces the electron gyrofrequency—but ion-cyclotron resonance might be accurately simulated. Although disadvantageous, it is precisely because SLPIC neglects fast motions that it can offer faster simulation. Importantly, in this paper we apply SLPIC to a steady-state problem in the sense that in the very beginning of breakdown, the space charge has negligible effect on the applied constant field; in the steady-state limit, we expect SLPIC to be essentially as accurate as PIC.

Reiterating an important point: the flux weight $w_{j,p} = w_p(t)/\beta(v_p(t))$ of a SLPIC macroparticle is a constant in time, and this ensures that the flux density is divergenceless in the steady-state limit. For example, if a particle receives an abrupt kick—whether from the electric field or from a collision—its $w_{j,p}$ remains constant. If the impulse increases the particle's speed v_p, its density weight w_p must change accordingly so that $w_p = \beta(v_p)w_{j,p}$. In Sec. III, we will see that the flux weight is conserved when simulating collisions involving SLPIC particles.

III. COLLISIONS IN SLPIC

Collisions can be simulated in PIC with a PIC-MCC algorithm;[11–13] the MCC algorithm can be used for SLPIC with just a few modifications: (1) using the flux weight $w_{j,p}$ instead of the density weight w_p (cf. §II B), and (2) considering the slowing of time when determining the collision rate. (Since most SLPIC simulations will have macroparticles with a distribution of $w_{j,p}$, the PIC-MCC algorithm must be able to handle macroparticles with different weights before modification for SLPIC.)

Before discussing collisions in SLPIC, we briefly review the pertinent aspects of collisions in PIC. In PIC-MCC, a binary macroparticle collision is not at all the same as a binary

collision between two physical particles. The collision between two macroparticles must statistically represent collisions between the two swarms of physical particles represented by the macroparticles. The spatial distribution of physical particles is usually assumed to be uniform within the same grid cell containing the macroparticle centers, regardless of the macroparticle shape S [cf. Eq. (7)], and zero outside that cell.

For concreteness, let us consider electron impact ionization in PIC-MCC, in which a primary electron collides with a primary neutral atom, resulting in the scattering of the primary electron, the neutral "becoming" an ion, and the "creation" of a secondary electron. (Although there are other kinds of collisions, e.g., elastic or excitation, between electrons and neutrals that do not ionize, we consider only ionizing collisions for simplicity.) A primary electron macroparticle p with weight w_{p0} and velocity $\mathbf{v}_{p0}$ collides with a neutral macroatom with w_{n0} and $\mathbf{v}_{n0}$. The primary electron macroparticle will be split into two parts: one part with weight w_{p1} represents uncollided electrons and the other with weight w'_{p1} represents scattered primary electrons (recoiling from having just ionized atoms). Similarly, w_{n1} is the weight of uncollided neutrals, and w_i the weight of "scattered" neutrals, which are really ions and are now subject to acceleration in the applied electric field. In addition, secondary electrons are created with weight w_{se}. In principle there could be many scattered/secondary macroparticles with different velocities, but for simplicity we will assume at most one per primary species. The uncollided particles retain the original velocities; the other particles must be assigned new velocities according to the ionization process: $\mathbf{v}'_{p1}$, $\mathbf{v}_i$, and $\mathbf{v}_{se}$. Conservation of subatomic particles requires $w_{p0} = w_{p1} + w'_{p1}$, $w_{n0} = w_{n1} + w_i$, and $w'_{p1} = w_i = w_{se} = W_c$, where W_c is the number of (physical-particle) collisions expected to take place when the two macroparticles "collide."

The expected number of collisions W_c within a time interval Δt in a cell volume ΔV between two arbitrary particle distributions, f_p (primary electrons) and f_n (neutrals) is

$$W_c = \Delta V \Delta t \int d^3\mathbf{v}_{p0} \int d^3\mathbf{v}_{n0}\, f_p(\mathbf{v}_{p0}) f_n(\mathbf{v}_{p0}) \sigma(\mathbf{v}_{n0} - \mathbf{v}_{p0}) \|\mathbf{v}_{n0} - \mathbf{v}_{p0}\| \tag{13}$$

where $\sigma(\mathbf{v}_r)$ is the collision cross-section, which depends on the relative velocity $\mathbf{v}_r$ between colliding particles. To calculate W_c between two PIC macroparticles in the same grid cell, we substitute the appropriate distributions, i.e., $f_p(\mathbf{v}) = (w_{p0}/\Delta V)\delta^3(\mathbf{v} - \mathbf{v}_{p0})$ and $f_n(\mathbf{v}) = (w_{n0}/\Delta V)\delta^3(\mathbf{v} - \mathbf{v}_{n0})$, ignoring the shape functions of the macroparticles, treating them as uniform in one cell and zero outside. The expected number of collisions between the two

macroparticles is then

$$W_c = \Delta V \frac{w_{p0}}{\Delta V} \frac{w_{n0}}{\Delta V} \sigma(\mathbf{v}_{n0} - \mathbf{v}_{p0}) \|\mathbf{v}_{n0} - \mathbf{v}_{p0}\| \Delta t. \tag{14}$$

This can be verified by considering a single electron moving through a swarm of atoms of density $n_{n0} = w_{n0}/\Delta V$ with velocity $\mathbf{v}_{n0}$. In the reference frame co-moving with the atoms, $d = \|\mathbf{v}_{n0} - \mathbf{v}_{p0}\| \Delta t$ is the distance traveled by the electron in Δt, and $\lambda_{\text{mfp}} = (n_{n0}\sigma)^{-1}$ is its mean free path. Therefore d/λ_{mfp} is the expected number of collisions that a single electron would experience (for small Δt such that $d/\lambda_{\text{mfp}} \ll 1$). The total number of electron collisions is then $W_c = w_{p0} d/\lambda_{\text{mfp}}$, which is identical to Eq. (14).

There are multiple Monte Carlo strategies for colliding pairs of macroparticles that all yield the same W_c *on average*. For example, if $W_c = 0.01 w_{p0}$, we could avoid splitting the primary macroparticles and creating small-weight scattered/secondary particles by ignoring 99% of colliding pairs, and colliding 1% with an enhanced $W_c' = w_{p0}$. We will not discuss this in any detail, because these strategies are identical for PIC and SLPIC. We also omit other practical details that need no modification for SLPIC, such as determining scattered and secondary macroparticle velocities according to given differential cross-sections, and dealing with exceptional cases such as $W_c > w_{p0}$ (indicating that Δt is too large). The only differences introduced by SLPIC are the determination of W_c and the resulting macroparticle weights.

As described in §II B, in SLPIC it is better to verify physical accuracy by considering particle fluxes rather than numbers or densities. Thus the basic thought-experiment for developing SLPIC collision algorithms is not the collision of two macroparticles, but of two narrow beams of macroparticles intersecting in a small volume; the particle fluxes into and out of that volume must satisfy the appropriate conservation laws. For electron impact ionization, this means that $w_{j,p0} = w_{j,p1} + w'_{j,p1}$, $w_{j,n0} = w_{j,n1} + w_{j,i}$, and $w'_{j,p1} = w_{j,i} = w_{j,se} = W_{j,c}$. Because flux weights w_j are constant in time [unlike density weights $w_p(t)$], this ensures that the fluxes of particles entering and exiting ΔV are appropriately perserved; i.e., the flux of unscattered plus scattered primaries leaving ΔV equals the flux of primaries entering ΔV, and the fluxes of secondary electrons and scattered primaries leaving ΔV are equal, etc.

The rule for determining the collision rate, $W_{j,c}/\Delta t$, is that SLPIC preserves the physical mean free path. For example, consider a steady-state, monoenergetic beam of particles in

the $+x$ direction, entering some scattering medium with mean free path λ_{mfp}. Scattering causes the density of original beam particles to decrease as $\sim \exp(-x/\lambda_{\text{mfp}})$. Because this is a steady-state, SLPIC should render this density profile accurately, requiring that SLPIC macroparticles experience the physical mean free path. In other words, SLPIC macroparticle must scatter with probability $d'/\lambda_{\text{mfp}} = \dot{x}\Delta t/\lambda_{\text{mfp}} = \beta(v)v\Delta t/\lambda_{\text{mfp}}$ within time Δt (in contrast, a physical particle would scatter with probability $d/\lambda_{\text{mfp}} = \dot{x}\Delta t/\lambda_{\text{mfp}} = v\Delta t/\lambda_{\text{mfp}}$). This can also be viewed as the reduction in collision rate due to the slowing down of time by a factor $\beta(v)$. The collision rate (in terms of fluxes) in ΔV must therefore be

$$\frac{W_{j,c}}{\Delta t} = \frac{w_{j,p0}}{\Delta t}\frac{d'}{\lambda_{\text{mfp}}} = \Delta V \frac{w_{j,p0}/\Delta t}{\Delta V} n_{n0}\sigma(\mathbf{v}_{n0} - \mathbf{v}_{p0})\|\mathbf{v}_{ni} - \mathbf{v}_{pi}\|\beta_p(v_{p0})\Delta t \tag{15}$$

where $\beta_p(v)$ is the speed-limiting function for electrons. This merely says that—if one were to measure fluxes into and out of ΔV—the flux of scattered electrons (i.e., $W_{j,c}/\Delta t$) would equal the flux of incident electrons (i.e., $w_{j,p0}/\Delta t$) times d'/λ_{mfp}.

The above expression involves the neutral density: $n_{n0} = w_{n0}/\Delta V = \beta_n(v_{n0})w_{j,n0}/\Delta V$. (In most applications, atoms will be slow and not speed-limited, hence $\beta_n \equiv 1$, but we want this treatment to extend to arbitrary collisions.) Thus:

$$\begin{aligned} W_{j,c} &= \Delta V \frac{w_{j,p0}}{\Delta V}\frac{\beta_n(v_{n0})w_{j,n0}}{\Delta V}\sigma(\mathbf{v}_{n0} - \mathbf{v}_{p0})\|\mathbf{v}_{n0} - \mathbf{v}_{p0}\|\beta_p(v_{p0})\Delta t \\ &= \Delta V \frac{w_{j,p0}}{\Delta V}\frac{w_{j,n0}}{\Delta V}\sigma(\mathbf{v}_{n0} - \mathbf{v}_{p0})\|\mathbf{v}_{n0} - \mathbf{v}_{p0}\|\beta_n(v_{n0})\beta_p(v_{p0})\Delta t. \end{aligned} \tag{16}$$

This expression (or rather $W_{j,c}/\Delta t$) yields the number of physical electrons scattered per Δt from the collision of particles represented by two SLPIC macroparticles. It is the same as Eq. (14), except: (1) flux weights w_j are used (not density weights w), and (2) the timestep is modified by a factor $\beta_n(v_{n0})\beta_p(v_{p0})$.

After calculating $W_{j,c}$ using Eq. (16), the rest of the collision algorithm proceeds as in PIC, except that where a weight was used in the PIC algorithm, the flux weight must be used in SLPIC. Once a macroparticle's flux weight $w_{j,p}$ is determined, its density weight is set by $w_p = \beta(v)w_{j,p}$.

IV. SIMULATION SETUP AND METHODOLOGY

We modelled an argon-filled parallel plate capacitor with one spatial dimension and three velocity dimensions. The electrodes were treated as particle-absorbing boundaries and the

gap-distance was fixed at $d = 1$ cm. We imposed a constant electric field $E = V/d$. The cell size Δx was set to a quarter of the mean free path. The number of cells for each simulation is given in Table I. We ran both SLPIC and PIC simulations. The setup parameters for the SLPIC and PIC simulations were identical except for the timestep. For PIC, we used a timestep given by $\Delta t = \Delta x/v_{\max}$, where $v_{\max} = \sqrt{2eV/m_e}$ was the maximum electron speed. For SLPIC, we used a timestep given by $\Delta t = \Delta x/v_0$, where $v_0 = v_{\max}\sqrt{m_e/m_{Ar}}$ was the electron speed limit imposed by $\beta(v_p)$ given in Eq. (11). This v_0 was chosen because it provided the maximum speed up without affecting the ion motion. The large mass difference between electrons and argon ions resulted in the SLPIC timestep being 270 times larger than the PIC timestep. Therefore, the SLPIC simulations required 270 times fewer timesteps than PIC. The number of timesteps for each simulation is given in Table I.

To initiate the discharge we injected 100 electrons at the cathode in the first timestep with zero initial velocity. We simulated five types of electron-neutral collisions: ionization, elastic collisions, and three excitations. Many different types of collisions occur between electrons and argon, but we simulated only the five most likely. The other collisions had negligible cross-sections in the relevant electron energy range. The cross-sections were extracted from the Biagi-v7.1 database[26] and Phelps database[21] on www.lxcat.net on August 15, 2020. We did not include any ion collisions. Although ion collisions could increase V_b, γ_{se} depends only weakly on the ion energy, so the effect of these collisions is negligible. For excitation collisions, the electrons were scattered isotropically with an energy reduced by the excitation threshold. For elastic collisions, the electrons were scattered according to the Vahedi-Surendra algorithm.[27] For ionization collisions, the products were generated according to an algorithm developed by Kutasi and Donkó[19]. We used an energy-dependent model of secondary electron emission due to ion impact at the cathode.[9] Secondary electrons were emitted from the cathode with zero velocity with a probability given by Eq. (17) that depends weakly on the energy of the incident ion ϵ_i.

$$\gamma_{se} = \begin{cases} 0.09\left(\frac{\epsilon_i}{700\,\mathrm{eV}}\right)^{0.05}, & \epsilon_i < 700 \text{ eV} \\ 0.09\left(\frac{\epsilon_i}{700\,\mathrm{eV}}\right)^{0.72}, & \epsilon_i \geq 700 \text{ eV} \end{cases} \tag{17}$$

To determine whether the simulation voltage V was above or below V_b, we tracked the ion population in the simulation over 30 ion crossing times ($30\sqrt{2m_{Ar}d/eE}$). For $V < V_b$, the ion population decreases with time after an initial rise, eventually returning to zero. For

$V > V_b$, the ion population increases with time. This criterion is equivalent to the breakdown condition given by $\gamma_{se}(e^{\alpha d} - 1) > 1$. This method provided a clear binary classification method of simulation results. For each pressure, we performed a grid search in the vicinity of the experimental V_b. Once a simulation above and a simulation below breakdown were found, we bracketed the interval with the respective voltages. These brackets are indicated by the bars in Fig. 2.

The evolutions of the ion population for simulations above and below breakdown at 1 Torr cm are shown in Fig. 1, for both SLPIC and PIC. It can be seen that for $V > V_b$, the ion population increases with time, and for $V < V_b$, the ion population decreases with time after the initial rise. While SLPIC and PIC give the same results for V_b, the dynamics of the simulation are clearly different. The initial rise in ion population is steeper for PIC because the seed electrons cross the simulation domain and ionize the argon gas almost immediately, while in SLPIC, the speed-limited seed electrons cross the domain more slowly and therefore take longer to ionize the argon gas. The speed limiting also causes the ion population to evolve more smoothly as electrons and ions now have comparable velocities and oscillate in and out of the simulation domain at similar frequencies, but out of phase. SLPIC cannot capture the transient behavior of the particles since the speed-limiting approximation Eq. (5) is not satisfied in this regime. However, we do see convergence of the SLPIC dynamics to PIC as we increase the speed-limit.

We bracketed the breakdown voltage for 11 values of *pd* ranging from 0.3 to 300 Torr cm. We performed simulations with the Vorpal[28] code distributed in VSim-11. The simulations ran on the CORI supercomputer at the National Energy Research Scientific Computing Center (NERSC). The number of cores for each simulation is given in Table I.

V. SIMULATION RESULTS

SLPIC accurately computed the Paschen curve for argon over three orders of magnitude in *pd*, 0.3 to 300 Torr cm, agreeing with PIC over the range where PIC was feasible, 0.3 to 2 Torr cm. For each value of *pd*, we bracketed the breakdown voltage with one simulation above and below breakdown. The Paschen curve generated by the SLPIC and PIC simulations is shown along with Paschen's law and experimental data from Phelps and Petrovic[9] in Fig. 2. For the values of *pd* where PIC was run, PIC and SLPIC classified each simula-

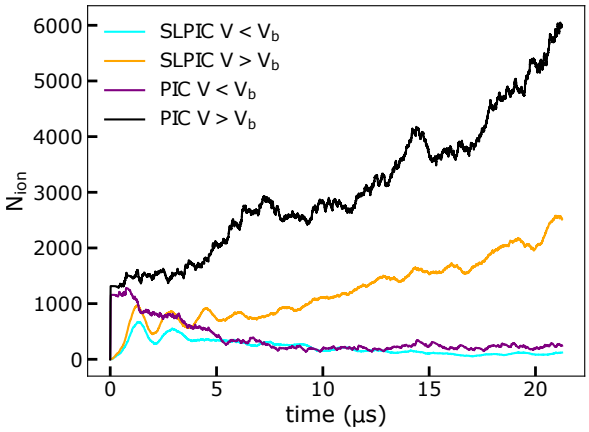

FIG. 1. The number of ions in the simulation domain as a function of time for the simulations with V nearest V_b at $pd = 1$ Torr cm. The SLPIC results are given for $V > V_b$ (orange) and $V < V_b$ (cyan). The PIC results are given for $V > V_b$ (black) and $V < V_b$ (purple). The number of ions in the simulation domain increases over time when $V > V_b$, and decreases when $V < V_b$.

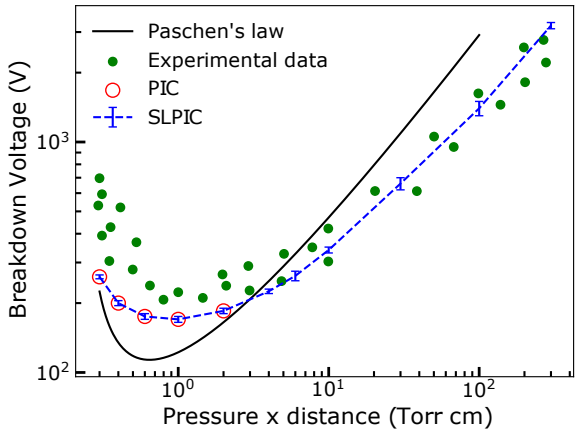

FIG. 2. The Paschen curve for argon. The experimental data (solid, green circles) was extracted from Phelps and Petrovic[9]. The values of A, B, and γ_{se} in Paschen's law (solid, black line) were taken from Lieberman and Lichtenberg[4]. The SLPIC results (blue bars connected by dashed, blue line) display the upper and lower brackets on the simulated breakdown voltage. The 5 lowest pressures (open, red circles) were also simulated using PIC, which yielded the same bounds as SLPIC.

TABLE I. Summary of setup parameters and performance for PIC and SLPIC simulations with $V \approx V_b$. The gap-distance was fixed at d =1 cm. The number of timesteps is given by N_T."Cores" refers to the number of CORI cores that the simulation was run on. The runtime T is given in core-hours. The speed-up is given by $T_{\mathrm{PIC}}/T_{\mathrm{SLPIC}}$.

p (Torr)	V (V)	Cells	$N_{T,\mathrm{SLPIC}}$	$N_{T,\mathrm{PIC}}$	Cores	T_{SLPIC}	T_{PIC}	Speed-up
0.3	260	66	4.0×10^3	1.1×10^6	4	0.26	11	42
0.4	195	88	5.3×10^3	1.4×10^6	4	0.27	14	52
0.6	175	131	7.9×10^3	2.1×10^6	4	0.49	91	187
1	170	219	1.3×10^4	3.5×10^6	4	0.78	113	146
2	185	437	2.6×10^4	7.1×10^6	4	1.8	91	51
4	225	874	5.2×10^4	1.4×10^7	4	6.3	1308 [a]	208
6	262	1310	7.9×10^4	N/A	4	9.2	N/A	N/A
10	340	2184	1.3×10^5	N/A	4	13	N/A	N/A
30	620	6550	3.9×10^5	N/A	128	105	N/A	N/A
100	1400	21832	1.3×10^6	N/A	128	771	N/A	N/A
300	3100	65494	3.9×10^6	N/A	128	6400 [a]	N/A	N/A

[a] time based on extrapolation due to time restrictions on CORI

tion identically, thus yielding the same brackets. Below 3 Torr cm, the simulated breakdown voltages were lower than those measured in experiment. At low *pd* the experiments approach the vacuum discharge regime where processes such as out-gassing, vacuum arc by burned cathode, and flashover become relevant to breakdown.[15,29] We did not simulate these effects, which may account for the discrepancy. Paschen's law, given in Eq. (1), is plotted using the values of $A = 11.5$ cm^{-1} Torr^{-1}, $B = 176$ V cm^{-1} Torr^{-1}, and $\gamma_{se} = 0.07$ taken from Lieberman and Lichtenberg[4]. Below 0.3 Torr cm, the simulation did not break down. This matches Paschen's law [Eq. (1)] which exhibits a singularity below 0.3 Torr cm.

SLPIC ran 40 to 200 times faster than PIC. The simulation parameters and performance for a subset of simulations run near the breakdown voltage are given in Table I. The speed-ups are the ratios of the PIC runtimes to the SLPIC runtimes and are given in the rightmost column. On average, the SLPIC simulations in Table I ran 116 times faster than PIC. Above

4 Torr cm, it became unfeasible to run PIC simulations. SLPIC enabled us to explore these higher pressures with a fully kinetic treatment.

VI. SUMMARY

Fast, accurate electrical discharge simulations are needed for the design of plasma processing equipment, sensitive microchips, and volatile chemical processing facilities. We have demonstrated that SLPIC is as accurate as PIC, but faster in predicting breakdown voltages of a gas-filled capacitor. SLPIC accurately computed the Paschen curve $V_b(pd)$ for argon for pd ranging from 0.3 to 300 Torr cm. SLPIC and PIC produced identical results, but SLPIC ran 40 to 200 times faster and extended the range of feasible simulations. In accurately computing the Paschen curve for argon, SLPIC has demonstrated that it can accurately model collisions, including electron-neutral ionization, electron-neutral elastic collisions, electron-neutral excitations, and ion-induced secondary electron emission. We expect that SLPIC will also be as accurate as PIC, but much faster, in simulating voltage breakdown in gases in more complicated 3D geometries as well as in the glow discharge regime where the discharge current alters the applied electric field.

ACKNOWLEDGMENTS

This work was supported by NSF grant PHY1707430 and by U.S. Department of Energy SBIR Phase II Award DE-SC0015762. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility operated under Contract No. DE-AC02-05CH11231.

AIP PUBLISHING DATA SHARING POLICY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1]J. Townsend, Nature **62**, 340 (1900).

[2]F. Paschen, Annalen der Physik **273**, 69 (1889).

[3]M. Sugiura, IEE Proceedings A (Science, Measurement and Technology) **140**, 443 (1993).

[4]M. A. Lieberman and A. J. Lichtenberg, *Principles of Plasma Discharges and Materials Processing* (John Wiley & Sons, 2005).

[5]A. Belasri, J. Boeuf, and L. Pitchford, Journal of Applied Physics **74**, 1553 (1993).

[6]X. Gao, J. J. Liou, W. Wong, and S. Vishwanathan, Solid-State Electronics **47**, 1105 (2003).

[7]T. Gillman and I. Le May, Engineering Failure Analysis **14**, 995 (2007).

[8]J. Ashwell, E. Cole, A. Pratt, and D. Sartorio, in *1958 IRE International Convention Record*, Vol. 5 (IEEE, 1966) pp. 72–77.

[9]A. Phelps and Z. L. Petrovic, Plasma Sources Science and Technology **8**, R21 (1999).

[10]J.-P. Boeuf and L. C. Pitchford, IEEE Transactions on Plasma Science **19**, 286 (1991).

[11]C. K. Birdsall, IEEE Transactions on Plasma Science **19**, 65 (1991).

[12]G. A. Bird, *Molecular gas dynamics and the direct simulation of gas flows* (Clarendon Press, Oxford, 1994).

[13]K. Nanbu, IEEE Transactions on plasma science **28**, 971 (2000).

[14]B. Bagheri, J. Teunissen, U. Ebert, M. M. Becker, S. Chen, O. Ducasse, O. Eichwald, D. Loffhagen, A. Luque, D. Mihailova, *et al.*, Plasma Sources Science and Technology **27**, 095002 (2018).

[15]G.-Y. Sun, B.-H. Guo, B.-P. Song, G.-Q. Su, H.-B. Mu, and G.-J. Zhang, Physics of Plasmas **25**, 063502 (2018).

[16]J. Adam, A. G. Serveniere, and A. Langdon, Journal of computational physics **47**, 229 (1982).

[17]G. R. Werner, T. G. Jenkins, A. M. Chap, and J. R. Cary, Physics of Plasmas **25**, 123512 (2018).

[18]M. Radmilović-Radjenović and B. Radjenović, Plasma Sources Science and Technology **15**, 1 (2005).

[19]K. Kutasi and Z. Donkó, Journal of Physics D: Applied Physics **33**, 1081 (2000).

[20]D. Mariotti, J. McLaughlin, and P. Maguire, Plasma Sources Science and Technology **13**, 207 (2004).

[21]C. Yamabe, S. Buckman, and A. Phelps, Physical Review A **27**, 1345 (1983).

[22]This approximation is subtly different from that in Werner *et al.*[17], but only when β has an explicit time-dependence—a regime that has not yet been investigated and is irrelevant for this paper.

[23]T. G. Jenkins, G. R. Werner, and J. R. Cary, Physics of Plasmas **28**, 062107 (2021).

[24]J. Teunissen and U. Ebert, Journal of Computational Physics **259**, 318 (2014).

[25]G. Lapenta and J. Brackbill, Computer Physics Communications **87**, 139 (1995).

[26]S. Biagi, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **421**, 234 (1999).

[27]V. Vahedi and M. Surendra, Computer Physics Communications **87**, 179 (1995).

[28]C. Nieter and J. R. Cary, Journal of Computational Physics **196**, 448 (2004).

[29]A. Descoeudres, Y. Levinsen, S. Calatroni, M. Taborelli, and W. Wuensch, Physical Review Special Topics-Accelerators and Beams **12**, 092001 (2009).