Adaptive Simulations Enable Computational Design of Electron Beam Processing of Nanomaterials with Supersonic Micro-Jet Precursor

Matthew R. Henry[a] and Andrei G. Fedorov[a,b,*]

[a]*George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA*
[b]*Parker H. Petit Institute for Bioengineering and Bioscience, Georgia Institute of Technology, Atlanta, GA 30332, USA*
**Corresponding Author: AGF@gatech.edu*

Abstract

Focused Electron Beam Induced Processing (FEBIP) is a powerful tool for the "direct-write" of nanomaterials with the possibility of atomistic control on suspended 2D material substrates. FEBIP capabilities have been significantly expanded by using a localized jet-based delivery of precursors, and especially when a thermally energized supersonic micro-jet enhances the delivery of mass flux along with controlling the far-from-equilibrium thermodynamic state of adsorbed adatoms. The possibilities of growing nanomaterials with "dialed-in" composition with ultra-high growth rates and aspect ratios up to 100:1 have been demonstrated using the supersonic micro-jet-FEBIP. Bringing this scientific discovery to the level of maturity required for practical applications in additive nanomanufacturing requires simulation tools that are capable of capturing the complex flow physics of micro-jet-substrate interactions that bridge a wide range of flow regimes from the high-density gas micro-jet expanding into the vacuum environment of FEBIP. To address this significant computational challenge, a new approach has been developed and described in this work for a multiscale adaptive DSMC (Direct Simulation Monte Carlo) algorithm to predict the micro-jet gas dynamics in an FEBIP environment, spanning the full range of flow regimes from low Knudsen (Kn) number O(0.01) continuum flow to high Kn of O(10) for the molecular flow within a unified computational framework. The fundamental principles of the adaptive DSMC algorithm are described, its viability as a DSMC technique is demonstrated, and computational improvements for the benchmark cases are discussed in the context of advancing 3D nanofabrication with FEBIP. Ultimately, combining the first principle simulations *via* adaptive DSMC with the complementary experimental data will enable the creation of the powerful CAD tools for *in silico* design and optimal operation of micro-jet-FEBIP.

Keywords: Direct Simulation Monte Carlo, Multi-Scale, Molecular Gas Dynamics, Computational Fluid Dynamics

1 Introduction

Additive manufacturing has revolutionized the field of materials synthesis and processing in the past decade with critical advances made possible due to development of model-based design and simulation tools based on underlying fundamental physico-chemical and thermodynamic principles [1]. Of recent interest is demonstration of multiscale simulation methods for in silico design of synthesis routes for atomically thin 2D materials with unique thermomechanical and chemical properties [2][3]. FEBIP is a promising path towards "direct-write" 3D nanofabrication where deposition, etching and doping (defect engineering) capabilities are enabled in the same processing environment down to atomic-level control, with the highest impact applications in processing of emerging of 2D materials and their composites for electronics and quantum communication devices [4]. Realization of this FEBIP promise demands the high-fidelity model-based control of deposition such that desired nanostructures may be directly fabricated from CAD designs [5]. The current state-of-the-art in FEBIP CAD approaches either approximate the precursor impingement rate as a uniform frequency with a uniform sticking probability [6] or use calibration curves for particular precursor molecules, geometric configurations, and flow rates [7]. This is adequate in basic (traditional) gas precursor delivery scenarios under quasi-equilibrium conditions; however, the delivery of precursor molecules to the surface, and the surface mobility of adsorbed precursor molecules are heavily influenced by the gas-surface interactions of more complex scenarios that open up a number of unique capabilities for nanomaterial processing in FEBIP [8][9][10]. Indeed, the intriguing possibilities for deposition of nanomaterials with on-the-fly controlled composition/purity [8][11] and a high deposition rate [10] have been recently demonstrated using the supersonic micro-jet delivery of the precursor with a high non-equilibrium state in the region of e-beam impingement.

Traditionally, in FEBIP gas injection in a molecular (non-interacting) flow regime has been the standard approach to supplying precursor deposition molecules, and a geometric impingement model was used to estimate the gas-phase precursor supply rate to the substrate [12]. DSMC methods were later implemented to improve the predictive accuracy of gas-surface impingement for gas injection in the molecular flow regime [13]. More recently, greater complexity of gas injection scenarios have been applied to manipulate nanoscale fabrication processes. Carrier gas was used to increase the precursor supply rate for low vapor pressure precursors [14]. Gas jets introduced from the continuum flow regime can enhance the precursor surface mobility [10] and modulate the adsorbed precursor thermodynamic state [15]. Coupled continuum flow supersonic jet inert (carrier) and oxygen gas with molecular flow precursor injection increase both the deposition rate and control on-the-fly material composition [11]. The capability to simulate these more complex flows is necessary to determine the non-uniform and transient distribution of surface adsorbed precursor molecules for the incorporation in FEBIP CAD tools [5]. Additionally, the ability to map jet-surface interactions is applicable to all diffusion-modulated nanofabrication methods, extending the utility of the simulation tools for material synthesis beyond FEBIP [15]. DSMC is the appropriate prediction tool for these

complex flows, as it is able to simulate multi-species gas flows [16] and capture the continuum, transitional, or molecular flow regimes [17].

2. Hybrid Adaptive Simulation for Micro-Jet-FEBIP CAD

The DSMC technique is predicated on the idea that a relatively small number of simulation particles can capture the physical behavior of an unmanageably large number of real particles. In order to accomplish this, all of the physical interactions that take place among individual particles are adapted to apply to simulation particles that represent a large number of real particles. The presence of multiple regimes implies significant variation of particle density from one location to another since the weighting factor, F, representing the ratio of the real molecules to the simulation particles is commonly set to a fixed value for all cells in the simulation domain. This, unfortunately, results in too few particles per a cell in the molecular flow regime for accurate results or too many particles per another cell in the continuous flow regime for computational efficiency.

Several techniques have been applied to circumvent this problem. The simplest approach is to accept the computational inefficiency and provide the necessary computer memory and time to run a simulation with an excessive number of particles. Computational time may be reduced for this method by applying different time-steps to domains of significantly different particle (molecule) densities in order to reduce the computational resources committed to the low-density regions [18]. This technique has further been adapted to allow each cell to have its own optimal time-step [19]. Typically, this would be combined with particle cloning (splitting a simulation particle into two identical particles when flowing from higher to lower density cells) [20][21]. Another alternative is to custom tailor the cell dimensions to increase in regions where particle densities decrease; this has been done, for example, to simulate flow through an expanding nozzle [22]. More recently it has become common to automate cell transitions between a Navier-Stokes continuum flow solver and DSMC [23][24][25][26][27]. The latter approach is particularly effective when the majority of the domain is in the continuum flow regime and only the minor part of the domain requires the DSMC approach. However, it does not address the significant variation in particle density that may occur within the regions of transitional and molecular flow regimes, and it is not appropriate for the predominately molecular flow domain involved in gas injection simulations for FEBIP. To enable the comprehensive simulations for this challenging case of FEBIP, we introduce an adaptive DSMC algorithm, which adjusts cell weighting factors according to cell populations. This allows DSMC simulations to seamlessly transition from the continuum flow regime at the nozzle to transitional and molecular flow regimes where the transition boundaries are not known *a priori* or are not readily defined. This approach aims to maintain appropriate simulation particle populations in each cell for both simulation accuracy and computational efficiency. Further, a particle seeding technique is introduced to correct aberrant simulation artifacts due to cloning. The result is a single unified physical simulation method that can be applied with no *a priori* knowledge of the boundaries between continuum, transitional, and molecular flow regimes using the same uniform

computational representation of flow physics throughout the domain. It establishes the fundamental foundation for advancing the rigorous and computationally efficient CAD tools of Micro-Jet-FEBIP and expanding its application using *in silico* design.

2.1 Direct Simulation Monte Carlo Fundamentals

The velocity distribution of the real particles in the isolated gas of Figure 1.a can be matched by a relatively small number of representative particles (Figure 1.b). The temperature and field velocity may be calculated from the velocity variance and average of the representative particles just as well as from the real particles, greatly reducing the information needed to perform a simulation. The number of real particles represented by a simulation particle is denoted by the weighting factor F.

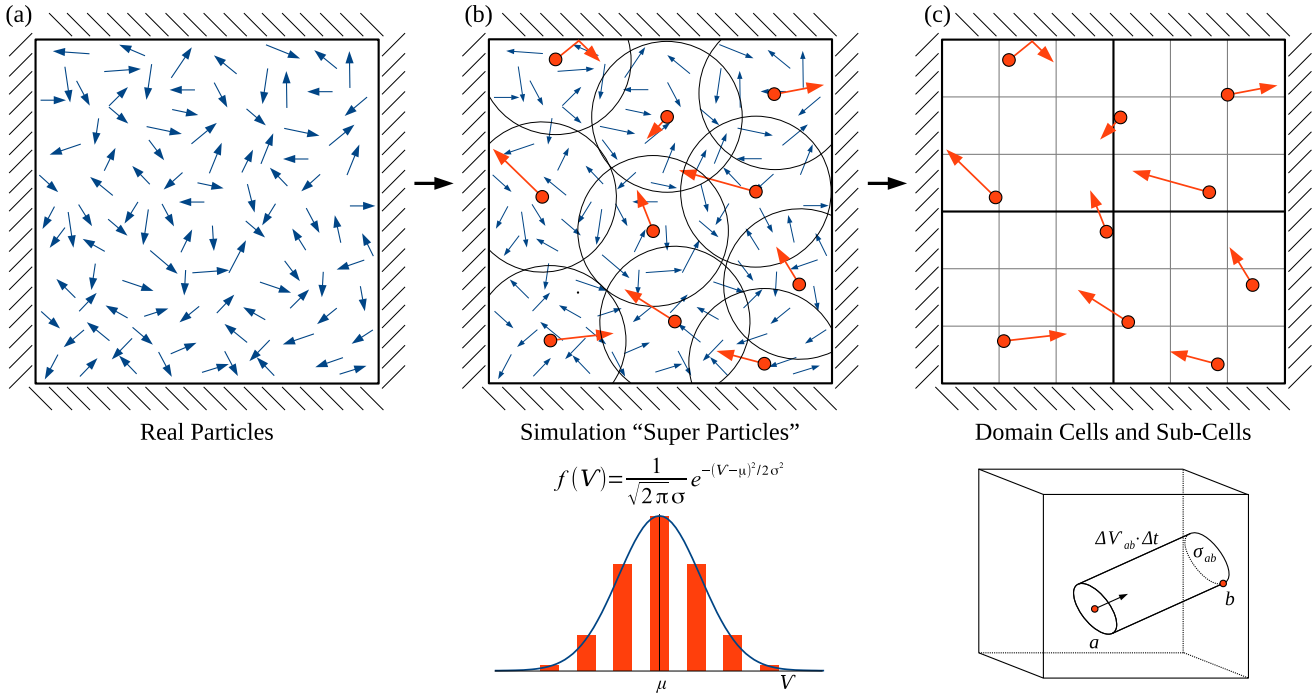

Figure 1*: a) This sample domain of a thermally isolated gas contains more real particles than are necessary to capture the relevant physical behavior of the gas and therefore pose a greater computational challenge than is necessary. b) A smaller number of representative particles are selected and sufficient to capture the same velocity distribution as the real particles. c) To simplify collision calculations, the domain is divided into cells and sub-cells. Collision events are chosen probabilistically based on collisional sweeping volume relative to sub-cell volume and the proximity of collisions is enforced by only allowing particles in the same sub-cell to collide.*

The second approximation of DSMC is the manner in which simulation particle collisions are implemented. Rather than calculating the position and time of particle intercepts, adjacent particles are made to collide with each other based on a locally calculated probability of collision. The domain is divided into cells and sub-cells, as indicated in Figure 1.c, and

representative particles are only considered to collide with each other if they are both within the same sub-cell. Based on the experience of previous DSMC developers, F should selected such that approximately 20 representative particles are in each a sub-cell (not as represented in Figure 1.c) [28][29], and the sub-cell dimensions must not exceed one-third of the local mean free path [17][29][30]. The simplest collision model is that of the Hard Sphere, which will be described by first considering real particles. In a given time step, Δt, two real particles in a sub-cell sweep out a collisional volume of $\sigma_{ab} \cdot (\Delta \mathrm{V}_{ab} \Delta t)$ where σ_{ab} is the cross-sectional area using the average of the two diameters, and ΔV_{ab} is the relative velocity of particle a to particle b. The time step should be chosen such that between time steps no particle may travel further than the nearest neighbor cell and no collisional volume is greater than the sub-cell volume. The collisional probability between these two real particles in a given time-step is then the collisional volume divided by the cell volume. The total number of real collisions is the sum of collisional probabilities between every pair of particles in the cell. Using N_R to represent the number of real particles in the cell, the total number of combinations is $(N_R^2 - N_R)/2$. (This can be visualized by taking a matrix of N_R rows and N_R columns. The number of combinations is N_R^2. A particle cannot collide with itself, so the diagonal is removed. And the upper-right triangle is a duplicate of the lower-left triangle, so the result is divided by two.) The total number of real collisions is then given by Equation 1. The number of simulation particles is $N_S = N_R/F$, the number of simulation collisions is $1/F$ times the number of real collisions, and $N_R^2 \gg N_R$, so the number of simulation collisions per sub-cell is given by Equation 2. The collision is then evaluated relative to the center of mass velocity with one particle randomly assigned a rebound zenith angle between 0 and π (where 0 is total rebound and π indicates no trajectory changes) and an azimuth angle between 0 and 2π. The other particle is assigned the resultant velocity that conserves energy and momentum.

$$\text{Real Collisions} = \frac{[(N_R^2 - N_R)/2]\langle \sigma_{ab} \Delta \mathrm{V}_{ab} \rangle \Delta t}{\forall_{\text{sub-cell}}} \tag{1}$$

$$\text{Simulation Collisions} \approx \frac{N_S^2 F \langle \sigma_{ab} \Delta \mathrm{V}_{ab} \rangle \Delta t}{2 \forall_{\text{sub-cell}}} \tag{2}$$

The Hard Sphere model does not, however, accurately represent real particle interactions. The surface of an atom, for example, is the result of its electron cloud exerting a repulsive force on other particles that are close enough to appreciably interact with it. As such, they are effectively soft spheres. When particles pass by each other at significant velocities, there is less time for the repulsive force to affect particle trajectories. Therefore, the Variable Hard Sphere [31][32] and the Generalized Hard Sphere [33] models were developed to adjust particle cross-sections based on relative velocity. Further, the non-rigid nature of a particle's surface causes the rebound angles to not be uniformly distributed as described; therefore, the Variable Soft Sphere [17][34] and the Generalized Soft Sphere [33] models use weighting factors in selecting rebound angles. The reader may also consider implementing the Virtual Sub-Cell collision pair selection method that is used to reduce

computational costs relative to random pair selection. The improvement is; however, marginal for three-dimensional simulations [35].

In summary, the main DSMC cycle is composed of three steps 1) Collisions: particle trajectories are adjusted based on probabilistic local sub-cell collisions, where the likelihood of a collision is based solely on collisional volume and not proximity. Since the representative particles are representative of particles throughout the cell, a single cell is effectively a single point, and the proximity nature of collisions is satisfied by the constraint of interacting only with other particles in the same sub-cell. 2) Motion: particles are moved on their trajectories for a single time step in which they may cross into new sub-cells or interact with a boundary. 3) Indexing: when particles move to a new sub-cell, their computational addresses must be updated. It is common to separate this from the motion module to facilitate parallel computing. To initiate an isolated simulation, as in the example given in Figure 1, it is necessary to specify the initial temperature, pressure, and velocity of each cell. The number of particles to place in each sub-cell is $N_S = \forall_{\text{sub-cell}} P/(F k_B T)$, where P is the specified pressure, and T is the specified temperature. Each axis of the particle's velocity is selected by a random number from the Gaussian distribution $f(\mathrm{V}) = exp\left[-(\mathrm{V}-\mu)^2/2\,\sigma^2\right]/\sqrt{2\pi}\,\sigma$, where μ is the specified mean velocity, $\sigma = R_U\,T/M_m$, R_U is the universal gas constant, and M_m is the molar mass.

If one of the boundary surfaces has an imposed gas flow across it, a permeable surface, then there is implicitly a fourth step to the cycle of adding boundary flux particles. Calculating the velocity distribution and the number of particles to add at such a boundary is non-trivial and is included here for a lack of a clear derivation in the existing literature. In considering the inlet flux across a boundary surface, the average velocity of particles moving in the "positive" direction is given by Equation 3, for which $\Phi(x) = 1 + erf\left[x/\sqrt{2}\right]/2$, and μ is not necessarily positive. The density of particles entering the domain is given by Equation 4 as the local particle density, n, times the fraction of particles moving in the "positive" direction. The inlet flux across the surface is the product of the average velocity and density of the inlet particles. This leads directly to Equation 5 for the number of particles to add to the surface of a boundary sub-cell at each time step. Of the particles moving in the "positive" direction, the probability of a particle crossing the surface is proportional to its velocity. Therefore, the velocity distribution is $g(\mathrm{V}) = \mathrm{V}f\,(\mathrm{V})/c$, in which c is the normalizing parameter $c = \mu \cdot [1 - \Phi(-\mu/\sigma)] + \sigma \cdot exp\,(-\mu^2/2\,\sigma^2)/\sqrt{2\pi}$. In this manner, the number of particles introduced at a permeable surface, and the orthogonal projection velocity distribution, are fully defined by the imposed pressure, temperature, and mean velocity at the boundary. The velocities for the two axes tangent to the surface are given by the baseline distribution $f(V)$.

$$\langle \mathrm{V}_P \rangle = \frac{\int_0^\infty \mathrm{V} f(\mathrm{V}) d\mathrm{V}}{\int_0^\infty f(\mathrm{V}) d\mathrm{V}} = \frac{\sigma e^{-\mu^2}/2\sigma^2}{\Phi(\mu/\sigma)\sqrt{2\pi}} + \mu \tag{3}$$

$$n_P = n \cdot \int_0^\infty f\,(\mathrm{V})d\mathrm{V} = \frac{P}{k_B T} \cdot \Phi\left(\frac{\mu}{\sigma}\right) \tag{4}$$

$$N_{boundary} = \frac{A_{\text{sub-cell}}\Delta t}{F}\frac{P}{k_B T}\left[\frac{\sigma e^{-\mu^2/2\sigma^2}}{\sqrt{2\pi}} + \mu \cdot \Phi\left(\frac{\mu}{\sigma}\right)\right] \tag{5}$$

The remaining boundary conditions are less complex. When a gas is in equilibrium across a free surface with a mean velocity of zero, the velocity distribution orthogonal to the boundary is $g(V)$, and the two tangential velocity distributions are still $f(V)$. The same constraints are applied for a particle rebounding from a diffuse non-adiabatic surface (see Supporting Information). For a diffuse adiabatic surface, the original velocity is preserved and the direction is selected based on equal probability for motion over a hemisphere above the surface. For a surface of symmetry, particles are specularly reflected back from the surface.

The results of the simulation are obtained by sampling the number of particles, the average velocity, and the velocity variance in the cells and sub-cells. According to the velocity distribution $f(V)$, the temperature of a particle species is related to the velocity variance along a single axis according to Equation 6, where m is the mass of an individual real particle, and N is the number of simulation particles that have been sampled. Pressure is determined by the number of particles, the temperature, and the volume. When multiple species are involved, the cell pressure is the sum of the partial pressures of each species, and the temperature and velocity are the averages of each species weighted by the relative population number density. Because there are typically 20 particles in each sub-cell, there are insufficient particles in a given time step for statistical calculations. If the simulation of interest is a steady-state simulation, then the same sub-cell of a converged simulation may be sampled over several time-steps to acquire an adequate sampling. To extract results from a transient simulation, it is necessary to collect a larger number of particles at each time step. The main purpose of designating sub-cells within cells is to provide a sufficient particle population in a cell to calculate the field properties of the cell (rather than the sub-cell), yet maintaining sufficiently few particles in each sub-cell for computational efficiency. (The other purpose is to increase the proximity of collision pairs [36] and therefore preserve flow rotation [37][38].) In this way, spatial resolution can be traded in exchange for temporal resolution with the same computational efficiency.

$$T_{x,i} = \frac{m_i}{k_B N_i^2}\left[N_i \sum^{N_i} u_i^2 - \left(\sum^{N_i} u_i\right)^2\right] \tag{6}$$

The DSMC technique is known to produce accurate results for molecular, transitional, or continuum flow regimes; however, it is typically applied to the problems involving only one of these flow regimes and requires special treatment under conditions in which all three flow regimes are realized within the same computational domain. The involvement of multiple regimes implies significant variation of particle density from one cell to another such that any choice for F necessarily results

in too few particles per sub-cell in the molecular flow regime for physical accuracy or too many particles per sub-cell in the continuous flow regime for computational efficiency.

2.2 Proposed Adaptive DSMC Algorithm

Here, we introduce an adaptive algorithm, which adjusts F for each sub-cell according to sub-cell populations, to allow DSMC simulations to seamlessly transition between flow regimes where the transition boundaries are not known *a priori* or are not readily defined. By allowing each sub-cell to have its own ratio between simulation particles and physically real particles, the simulation can capture the full range of flow regimes using a uniform computational representation of flow physics throughout the simulation domain.

When a particle moves from one sub-cell to another with a lower F factor, it must be split into two particles to conserve mass, as illustrated in Figure 2. When a particle moves into a sub-cell with a higher F factor, it must be managed in one of two ways to conserve mass. It may be put into a holding array for that sub-cell until a second particle arrives for it to merge with, or it may be subjected to a fifty percent chance of being either deleted or doubled in mass. The former strictly conserves mass, and the latter conserves mass on average. Because particles can only double or halve in mass, each sub-cell F factor must relate to all other factors by a power of two. The concern is how to accommodate these adjustments without distorting the velocity distribution; without adding or deducting net mass, energy, or momentum from the simulation; and without biasing the simulation.

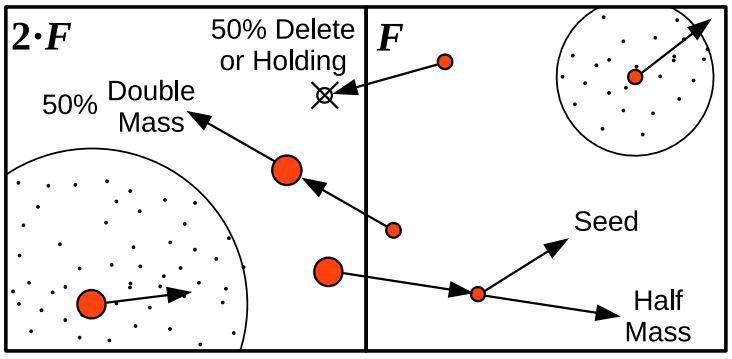

Figure 2*: A particle that moves to a sub-cell with a lower F is split and the resultant second particle needs to be seeded with velocity information. When particles moves to a sub-cell with a higher F, half of them are doubled in mass and half are either deleted or moved to a "Holding" array.*

When applying the merge strategy to particles that move to a sub-cell with a higher F factor or to sub-cells being adjusted to a higher F factor, only the velocity of one of the two merging particles can be preserved. Any combination of the velocities will distort the velocity distribution. The discarded particle velocity may be greater or lower, but the velocity distribution of the deleted particles, of the particles not deleted, and of the particles prior to merging, are locally all the same

distribution. Therefore, the velocity distribution is unchanged by discarding the velocity information of half of all merged particles. The potential error that a merging event introduces is equal to the resolution error of the sub-cell with the higher factor (half of a particle), so the important aspect of this error is that it not be directionally biased towards higher or lower energy. Because the velocity data of half of all merged particles is discarded, none of their information persists beyond the merge event. It is therefore expedient to use the deletion strategy instead of the holding strategy.

When a particle is split due to entering a sub-cell with a lower F factor, or when its sub-cell is being adjusted to a lower F factor, one particle adopts the original velocity information, but new information must be generated for the second particle. This information cannot be taken from previously discarded information of merged particles because there is no one-to-one relationship between merging and splitting events. The conventional strategy, particle cloning, is to assign the same velocity information to the second particle [20][21]. (This strategy is also used when transitioning to regions of larger cell size that have the same particle density, but do not require the same spatial resolution [17].) Particle cloning should be minimized or avoided if possible due to statistical error. Specifically, the cloned particle increases the number of particles used to calculate the velocity variance, but it has zero variance relative to the original particle. As a result, particle cloning artificially reduces the gas temperature. The problem is resolved by using the most current temperature and velocity sampling information from the cell to seed the second particle's velocity information. This approach is readily applicable to steady-state simulations; however, transient simulations should account for the fact that velocity information seeded into the simulation is generated from sampling a prior simulation time state. The described method is realized as a custom software suite in C++ compiled by GCC and implemented with OpenMP.

The statistical error introduced by particle cloning is illustrated in Figure 3 for a sample 1D scenario in which an argon gas jet is introduced at $40Pa$, $200K$, 300 m/s, and Mach 1.14 then exits to vacuum at $100\mu m$. The first five rows of cubic cells ($1\mu m^3$ each) along the flow axis are assigned the weighting factor of 1024. A drop in the weighting factor is imposed upon each subsequent row of cells, as shown in Figure 3.b, until the eleventh position. The velocity distribution of this cell is sampled for 14,000 distinct velocities orthogonal to the mean flow velocity (local thermal equilibrium is lost throughout the 1D expansion), and the particle cloning and particle seeding methods are compared in Figure 3.c with the expected Gaussian distribution for argon gas at $200K$. In principle, the DSMC algorithm is not designed to capture the true velocity distribution of gas particles, but rather to discretize it with an adequate number of simulation particles. However, when a particle's velocity is duplicated in particle cloning, its particular velocity becomes over-represented and the velocity distribution becomes artificially distorted for the given number of particles. Particle seeding averts this problem by introducing velocity information spawned from the local cell temperature and mean velocity. The distribution discretization generated by particle seeding in Figure 3.c is consistent with the true discretization, based on local temperature and average velocity that must occur for 14,000

random velocities. Since the local cell properties in this case are determined equally from upstream information and old information (due to seeding), the particle statistics will converge towards the true cell properties. To quantitatively compare cloning to seeding, the Root-Mean-Square Deviation (RMSD) of each velocity distribution relative to the canonical Gaussian distribution is calculated for all bins. The RMSD for cloning is 6.4 times greater than for seeding, and this is approximately constant for all bin sizes from 1 to 90*m/s*.

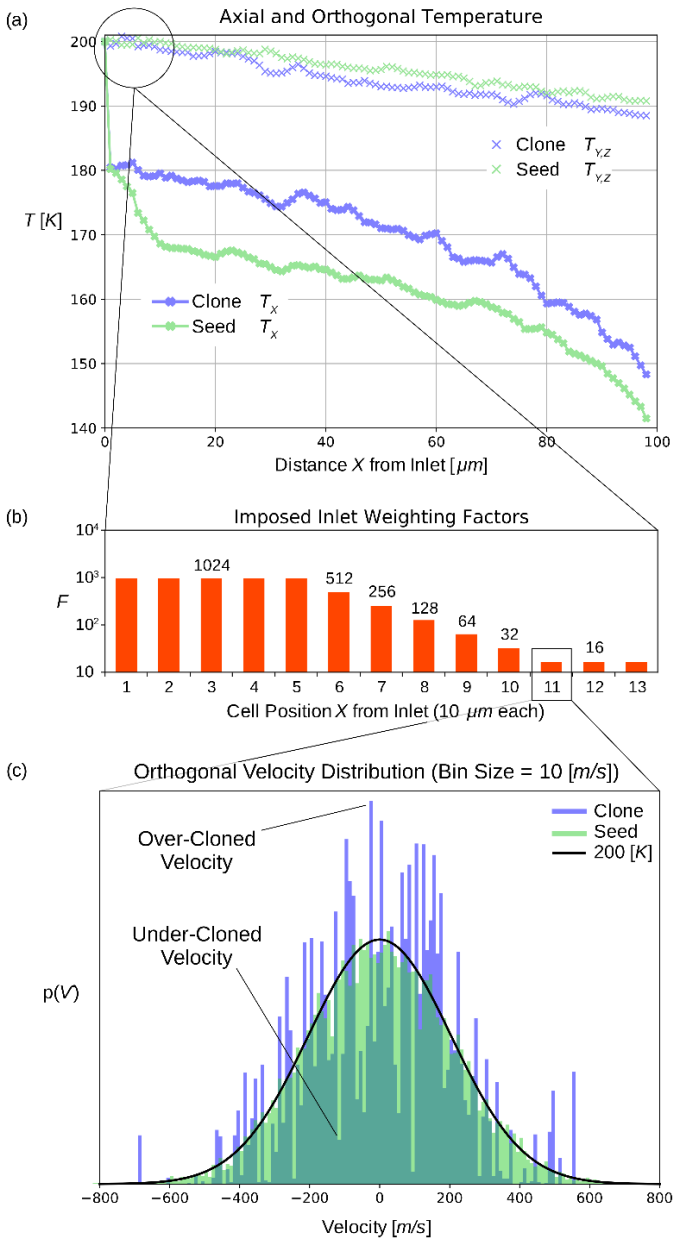

***Figure 3**: (a) A 1D argon gas isentropically expands from inlet conditions to an imposed vacuum at 100μm. (b) Weighting factors are imposed to produce a controlled particle-splitting scenario. (c) Histograms of 1000 sampled particles*

from each flow (cloning and seeding) show that particle seeding significantly superior in capturing the Gaussian velocity distribution.

An excessive number of particles in a sub-cell diminishes computational efficiency and insufficient particles compromises simulation accuracy. Therefore, sub-cells must double or halve their individual weighting factors to maintain a local particle population close to the specified target value, N_T. Three options for the adjustment threshold are presented in Figure 4. One option is to adjust the weighting factor only when the particle population exceeds double or drops below half the target population. This option moves the particle population directly to the target value; however, it also produces the widest span of particle populations. A second option is to adjust the weighting factor when the local population deviates from the target value by a multiple of $\sqrt{2}$. This option adjusts the local cell population from one bounding threshold to the other bounding threshold, which, due to random population fluctuations, will persistently produce frequent cell adjustments. When a sub-cell merges its particles to double its weighting factor, it purges information from the system. When a sub-cell splits its particles to halve its weighting factor, it introduces seeded information. Therefore the frequency of sub-cell weighting factor adjustments must be minimized. The compromise threshold used to minimize the population span and the adjustment frequency is therefore $2^{0.75} = 1.682$. If cell adjustment would result in a neighboring sub-cell weighting factor differing by more than a multiple of two, then it is not allowed. The cells should adjust up or down together.

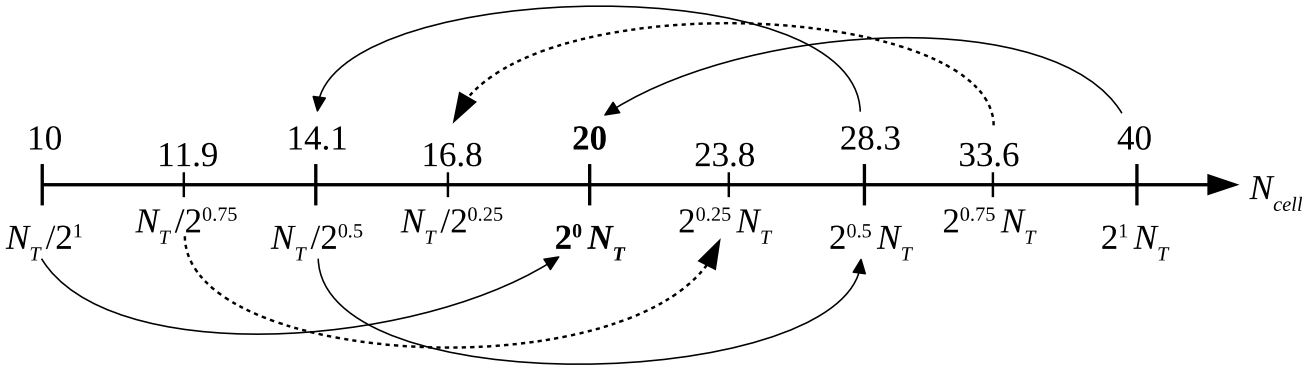

Figure 4*: Cell population thresholds must be determined to trigger weighting factor (F) adjustments. Three thresholds are considered as multiples of the target cell population (N_T). The optimal threshold multiple to minimize the population range and the update frequency is $2^{0.75}$=1.682.*

3. Method Verification and Application to Micro-Jet-FEBIP

3.1 Adaptive DSMC Benchmark Studies

For methodology validation in the transitional and molecular flow regimes, we have applied the adaptive DSMC algorithm to the benchmark problem of an argon gas jet expansion into a vacuum. The inlet conditions are a plug flow in the transitional flow regime at a Mach number of 3 and a Reynolds number of at least 25. (Additional configuration details for

all simulation cases are provided in Supplementary Information.) Previously, the problem was solved by treating the transitional and molecular regime flows as a continuum inviscid flow, and the solution was compared to the Method-of-Characteristics (MOC) solution to the Bhatnagar Gross Krook (BGK) approximation of the Boltzmann Transport Equation [39]. The adaptive DSMC simulation uses the hard sphere collision model with an inlet Knudsen numbers of 0.1 and 0.01. Horizontal and vertical symmetry are applied to reduce the axially symmetric simulation to one-quarter of the total 3D domain. Local density predictions are compared in Figure 5. The normalized density values predicted by the MOC at the flow centerline are compared to the adaptive DSMC predictions yielding Root-Mean-Square Deviations (RMSD) of 0.112 and 0.118 for inlet Knudsen numbers 0.1 and 0.01, respectively.

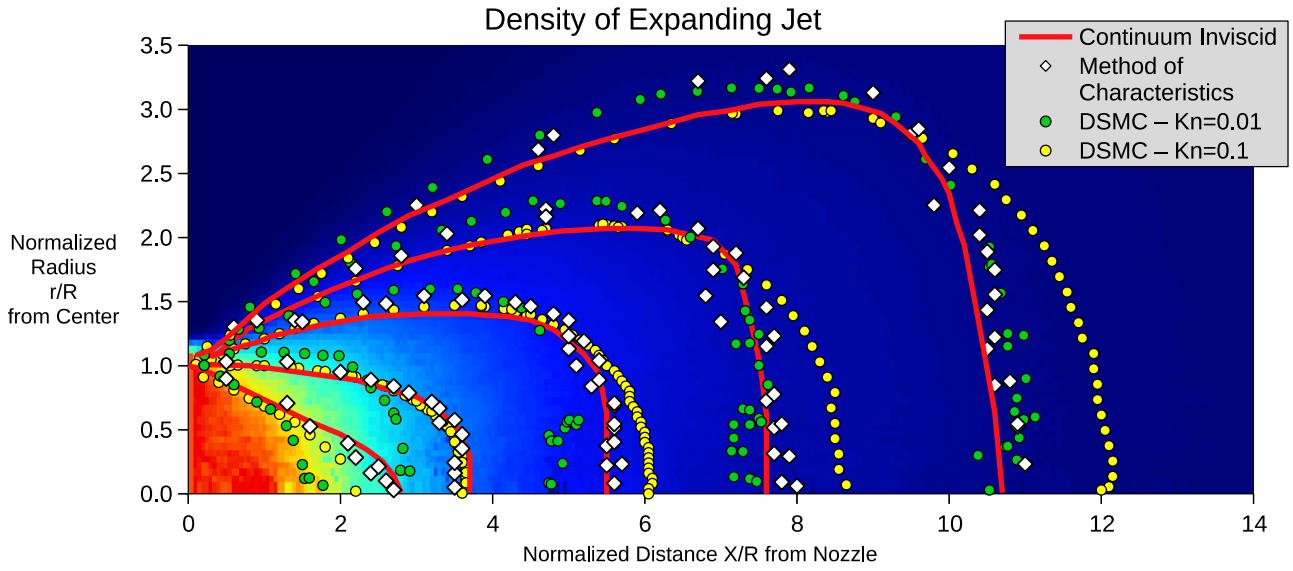

Figure 5*: Adaptive DSMC results are compared to a continuum inviscid calculation and a BGK approximation to a supersonic argon gas jet, which expands from the transitional flow regime to the molecular flow regime. The DSMC results for both inlet Knudsen numbers 0.1 and 0.01 align well with established results for relative density contour lines of 0.1, 0.05, 0.02, 0.01, and 0.005 (normalized by the stagnation density). The normalized density values predicted by the MOC at the flow centerline are compared to the DSMC predictions yielding Root-Mean-Square Deviations (RMSD) of 0.112 and 0.118 for inlet Knudsen numbers 0.1 and 0.01, respectively, which normalize by the MOC data ranges to 0.555 and 0.585. The figure background color shows the gradient that is indicative of the density distribution based on the DSMC simulation with inlet Kn=0.1.*

The agreement between the continuum inviscid calculation, the MOC solution to the BGK approximation, and the adaptive DSMC algorithm is expected in the rarefied transitional and molecular flow regimes in which viscosity is negligible. When the inlet Knudsen number is reduced to 0.01, the expansion of the gas jet from the continuum flow regime differs

slightly from the rarefied flow. As one would expect, including viscous effects would decrease the penetration depth of the jet due to an increased viscous momentum dissipation and this is well captured by the adaptive DSMC.

The total mass and energy entering and leaving the entire simulation domain, and those entering and leaving the rectangular sub-domain box indicated in Figure 6 are recorded to ensure global and local conservation, respectively. The global percent difference in mass and energy are 0.33% and 3.1%. Locally, they are 0.95% and 0.70%. Conservation data for other simulations are available in Supplementary Information.

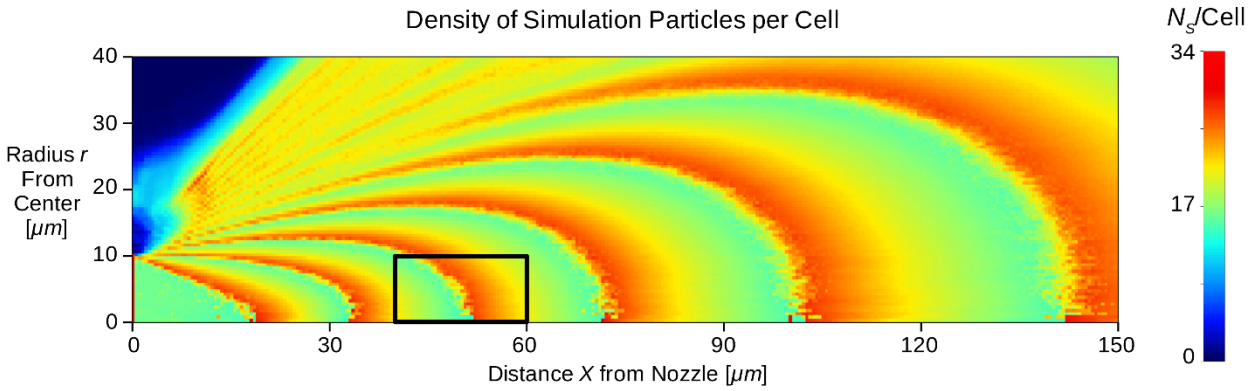

Figure 6*: The adaptive DSMC algorithm ensures that an appropriate number of simulation particles is used throughout the domain regardless of particle density. The adjustment of simulation particles per cell resulting from the adjustable per-cell weighting factors (F) is shown as a color map for this axially symmetric simulation for which the target number (N_T) is 20. A rectangular box with one edge on the center axis defines the boundaries to monitor local conservation.*

When a gas jet expands into a vacuum, it loses local thermal equilibrium. For argon gas jet simulations with inlet Knudsen number 0.1 (slip flow) and 0.01 (continuous flow), the normalized temperatures are plotted along the centerline axis in Figure 7. The axial temperature normalized by inlet temperature is T_x, and the normalized orthogonal component of temperature is T_r. The RMSD of T_r and T_x relative to data predicted by the MOC are 4.94×10^{-4} and 3.98×10^{-4}, which normalize by the MOC data ranges to 2.11×10^{-3} and 2.66×10^{-3}. For the inlet Knudsen number of 0.1, the axial and orthogonal components of temperatures are significantly different from each other and exactly match the predictions provided by the MOC solution. However, the lower Knudsen number simulation yields different results, which are expected due to a greater region of higher gas density in which local thermal equilibrium prevails. In both simulation cases, the axial and orthogonal temperatures distinctly separate when the Knudsen number exceeds 0.1 (flow entering the transitional regime).

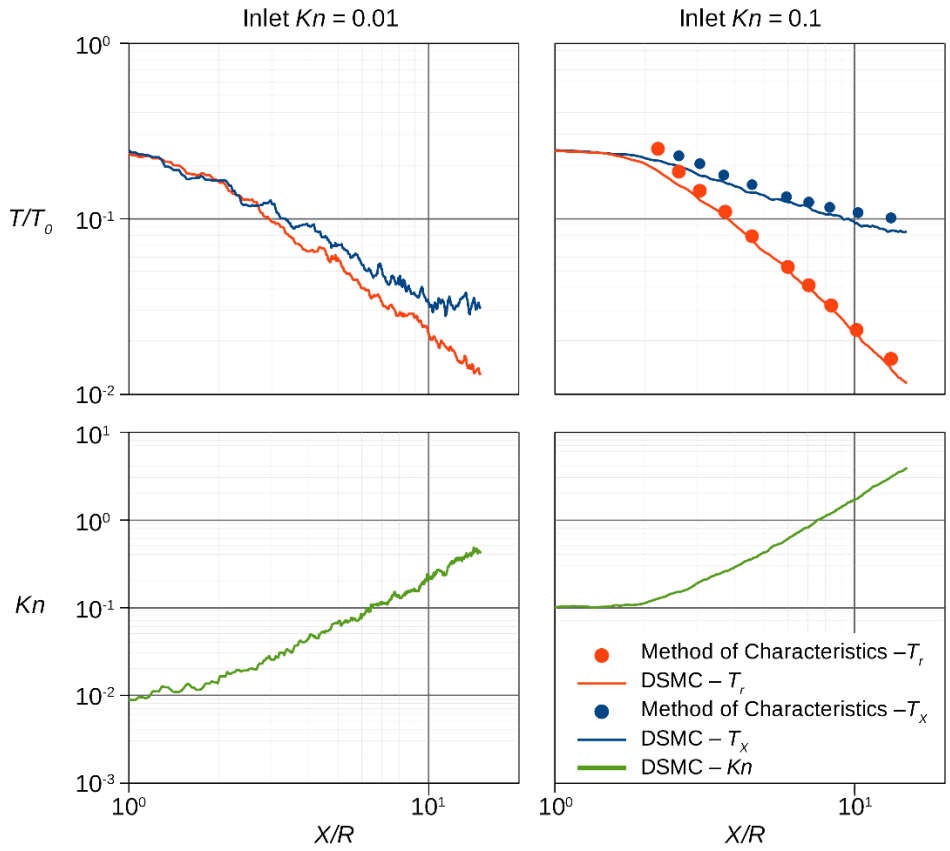

***Figure** 7: The axial and orthogonal temperature components normalized by the inlet temperature and the local Knudsen number are plotted along the centerline of each argon gas jet simulation. The RMSD of T_r and T_x relative to data predicted by the MOC are 4.94×10^{-4} and 3.98×10^{-4}, which normalize by the MOC data ranges to 2.11×10^{-3} and 2.66×10^{-3}.*

To verify the adaptive DSMC algorithm in the continuum flow regime, it is applied to reproduce the expected temperature variation of argon gas in plane Couette flow accounting for volumetric heat generation due to viscous heat dissipation. The periodic flow is potentially the most challenging validation case for application of variable weighting factor schemes, in which errors may amplify upon "positive" (additive) feedback loop. Referring to Figure 8, both surfaces are maintained at $T_{Wall} = 273K$, the bottom and top surfaces are moving in opposite directions at $U_{Wall} = 100\,m/s$, the gap is $H = 1m$, the viscosity of argon is $\mu = 2.16 \times 10^{-5}\,kg/m\,s$, the thermal conductivity of argon is $k = 1.772 \times 10^{-2}\,W/m\,K$, and the average density is 7.04×10^{-6} kg/m^3. Using the hard sphere Van der Waals diameter, the Knudsen number is Kn~0.01. For this configuration, the analytical solution for the temperature profile is given by Equation 7 [29]. With reference to Figure 8, the entire simulation domain is divided into two subdomains with distinct imposed weighting factors (5.78×10^{10} in the core and 1.16×10^{11} near the upper and lower walls). This forced assignment of weighting factors tests the proposed merging and splitting algorithm in handling the periodic flow. The simulation has an RMSD 0f 0.232 relative to the theoretical prediction,

which normalizes by the theoretical range to 0.0340. This is compared to the same simulation with a single uniform weighting factor, which has an RMSD of 0.013 which normalizes to 0.0022.

$$T = T_{Wall} + \frac{2\mu U_{Wall}^2}{k}\left(\frac{1}{2Kn+1}\right)^2\left[\frac{z}{H} - \left(\frac{z}{H}\right)^2 + \frac{2m}{2c_P m - k_B}\frac{Knk_B}{\mu}\right] \quad (7)$$

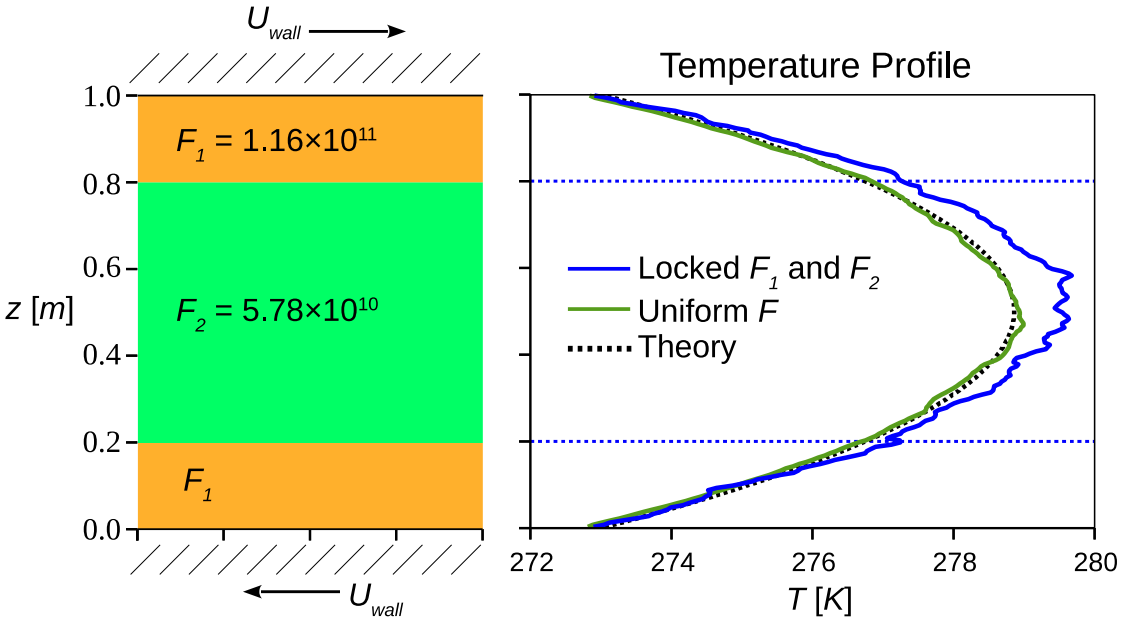

***Figure 8**: Periodic Couette flow with imposed changes in weighting factors near the upper and lower walls is used as a stress test for the adaptive DSMC algorithm. The simulation has an RMSD 0f 0.232 relative to the theoretical prediction, which normalizes by the theoretical range to 0.0340. This is compared to the same simulation with a single uniform weighting factor, which has an RMSD of 0.013 which normalizes to 0.0022.*

3.2 Application of Adaptive DSMC to Complex FEBIP Flows

The verification case studies demonstrated the applicability of the adaptive DSMC algorithm to the molecular and continuum flow regimes. Complex gas injection systems enabling advanced modes of FEBIP involve flow domains of different density where different F numbers apply in complex flow structures (from continuum to molecular and back to continuum) that are not known *a priori*. One such compelling case of gas jet assisted FEBIP of nanomaterials, in which the experimentally observed beneficial effects produced by a continuum flow micro-jet can only be understood and fully exploited with full simulations of the gas flow [40][8][41]. Figure 9 shows the adaptive DSMC algorithm is able to seamlessly adapt to a complex flow, revealing – for argon injected at sonic velocity and 1.3×10^{-7} *kg/s* via a 75*μm* nozzle – a stagnation pocket forming at the solid surface. At the inlet, the average temperature, velocity, and pressure are set to 222.4 *K*, 277.3 *m/s*, and 4766 *Pa*, respectively. The local Knudsen number, based on the inner diameter, indicates that the complexity of flow regimes developed in the domain preclude any *a priori* designation of regions for distinct computational strategies. A clear distinction

is revealed between the free isentropic expansion of the gas jet and the region of the jet interaction with the surface, which emphasize the importance of being able to simulate the domains of widely different molecular density (from molecular to fully continuous) within a uniform computational framework.

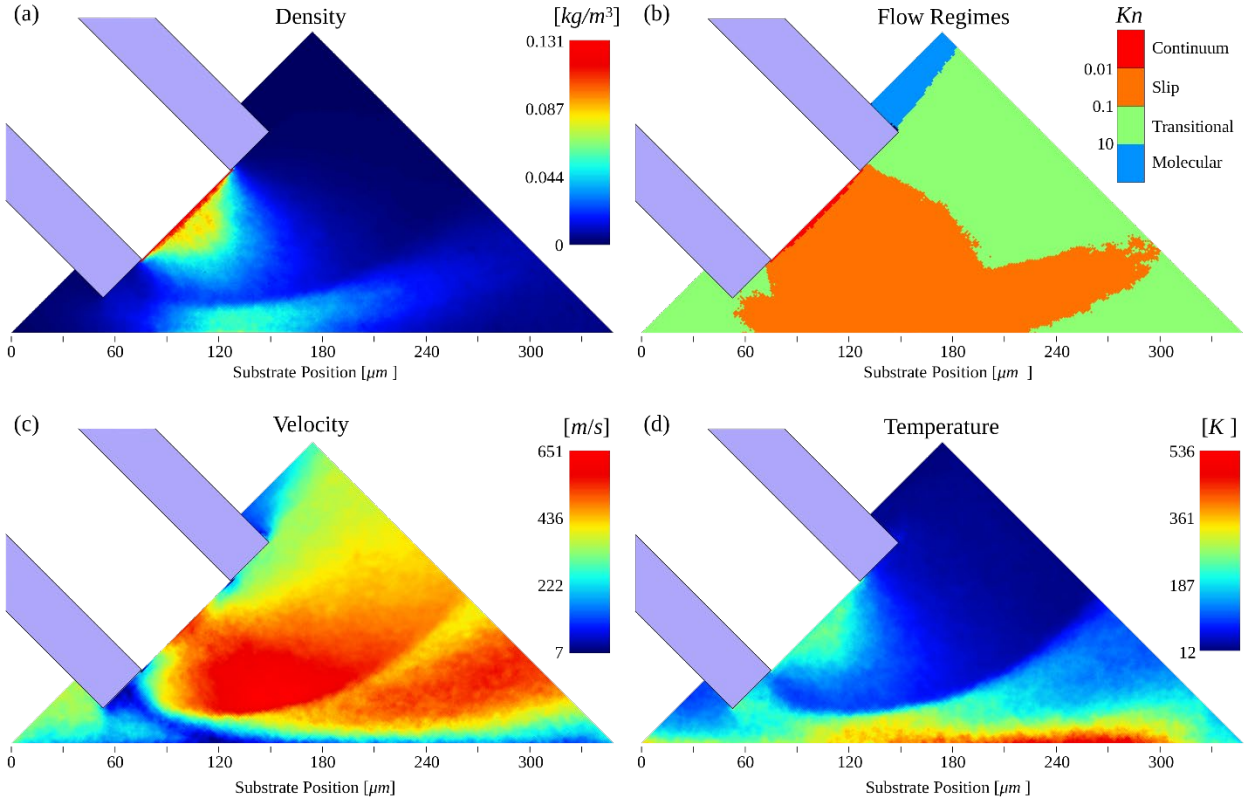

***Figure 9**: Adaptive DSMC results for a 1.3×10^{-7} kg/s argon continuum gas jet emanating at sonic choked flow velocity from a 75 μm nozzle to vacuum, which isentropically expands into the molecular flow regime before interacting with a solid surface. (a) The density distribution reveals a stagnation pocket on the solid surface spanning a range of approximately 60 μm. (b) The local Knudsen number, based on the inner diameter, indicates that the complexity of flow regimes developed in the domain preclude any a priori designation of regions for distinct computational strategies. (c,d) A clear distinction is revealed between the free isentropic expansion of the gas jet and the region of jet interaction with the surface, which emphasize the importance of being able to simulate the domains of widely different molecular density (from molecular to fully continuous) within a uniform computational framework.*

Prediction of nanomaterial growth in FEBIP is defined by impingement characteristics of the gas-surface interactions, as illustrated in Figure 10. The spatial distribution of per-particle impingement energy is critical to determining the local molecular sticking coefficient for precursor molecules entrained in the gas jet. Depending on surface roughness, the sticking coefficient may depend on the total or the orthogonal impingement energy [42]. The sticking coefficient, combined with the spatial distribution of the impingement rate, reveal the rate of molecular precursor supply to the solid surface. Finally,

the local velocity distribution, combined with impingement rate, are used to calculate the local spatially resolved surface temperature, which is validated by comparison to local measurement of the substrate temperature map *via* micro resistance thermal detectors [15]. The surface temperature strongly influences the diffusion rate of surface adsorbed precursor molecules, is a critical controlling factor in additive nanofabrication using FEBIP and other surface molecular assembly techniques [40][43].

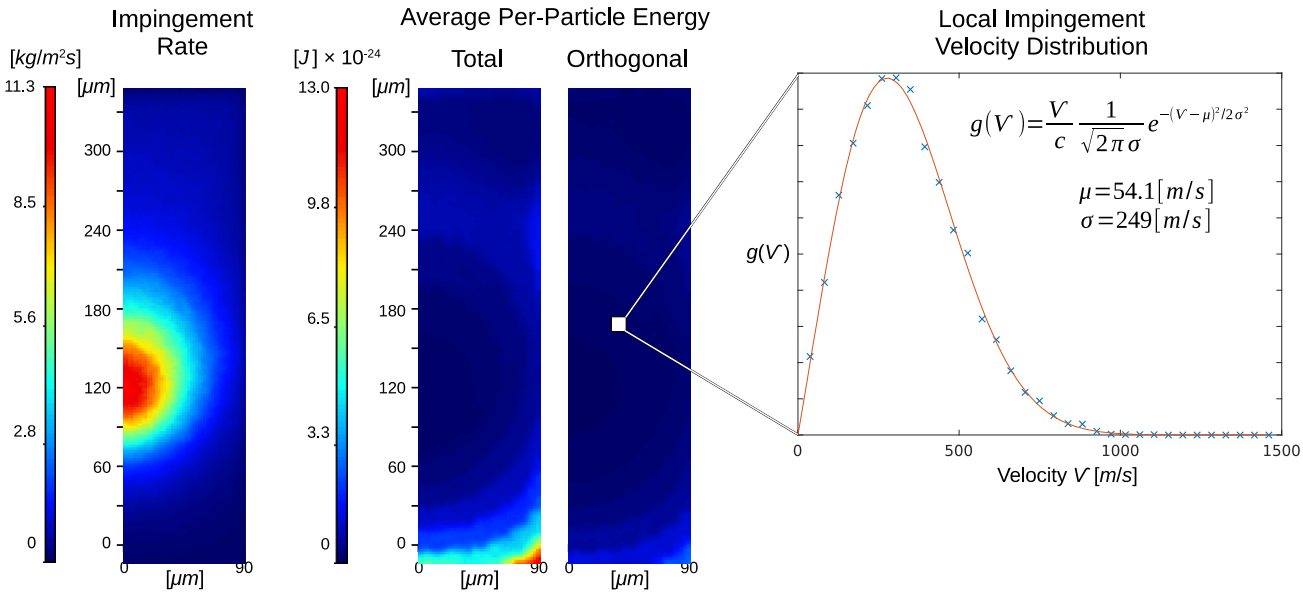

***Figure 10**: The DSMC algorithm predicts the impingement distribution, the impingement energy, and the localized impingement velocity distribution (see Fig. 8 caption for conditions). This provides a comprehensive set of information on how the gas jet interacts with the surface and contributes to FEBIP of nanomaterials.*

4 Conclusions

The new adaptive algorithm developed for DSMC enables computationally efficient simulation of multi-scale gas flows underlying the new capabilities for "direct-write" FEBIP of nanomaterials ultimately aiming towards the development of CAD toolboxes. Central to the adaptive algorithm are 1) seeding/splitting simulation particles with the current cell property data to prevent data aliasing and 2) auto-adjusting local weighting factors (ratio of actual to simulation particles) to maintain a sufficient and not excessive number of simulation particles in each cell. For adaptive DSMC, the trade-off between temporal and spatial resolution is the same fundamental trade-off faced by standard DSMC methods. Importantly, the adaptive algorithm is not affected by the collision model used and has no impact on any other aspects of the computational procedure, so it should be readily adaptable as a "drop-in" tool in any DSMC software package. We have demonstrated the viability of the technique with the hard sphere collision model for each flow regime of interest by comparing results generated with the adaptive DSMC algorithm with literature data for benchmark cases. The adaptive DSMC algorithm can be directly

incorporated into CAD software for 3D nanofabrication with FEBIP, or may be used to generate the heuristic curves for common gas injection configurations or data sets for machine learning (ML) models of the process, which are difficult and expensive to produce through experiments.

ACKNOWLEDGEMENTS

This work was supported by the U.S. Department of Energy (DOE), Office of Science, Basic Energy Sciences (BES), under Award #DE-SC0010729.

DATA AVAILABILITY

The authors declare that the data supporting the findings of this study are available within the paper and its supplementary information files.

COMPETING INTERESTS

The authors declare no competing financial or non-financial interests.

AUTHOR CONTRIBUTIONS

A. G. Fedorov served in an advisory role for all research reported in this article, including problem identification, formulation and applications.

M. R. Henry developed and implemented the multi-domain adaptive simulation algorithm as described in this article, and performed all simulation/validation studies.

REFERENCES

[1] E. Giessen, P. A. Schultz, N. Bertin, V. V. Bulatov, W. Cai, G. Csányi, S. M. Foiles, M. G. D. Geers, C. González, M. Hütter, W. K. Kim, D. M. Kochmann, J. LLorca, A. E. Mattsson, J. Rottler, A. Shluger, R. B. Sills, I. Steinbach, A. Strachan, E. B. Tadmor, Roadmap on Multiscale Materials Modeling, Modelling Simul. Mater. Sci. Eng. 28 (2020) 043001.

[2] K. Momeni, Y. Ji, Y. Wang, S. Paul, S. Neshani, D. E. Yilmaz, Y. K. Shin, D. Zhang, J. Jiang, H. S. Park, S. Sinnott, A. Duin, V. Crespi, L. Chen, Multiscale Computational Understanding and Growth of 2D Materials: A Review, NPJ Comput. Mater. 6 (2020) 22.

[3] S. Paul, K. Momeni, Mechanochemistry of Stable Diamane and Atomically Thin Diamond Films Synthesis from Bi- and Multilayer Graphene: A Computational Study, J. Phys. Chem. C 123 (2019) 15751−15760.

[4] A. G. Fedorov, S. Kim, M. R. Henry, D. Kulkarni, V. V. Tsukruk, Focused Electron Beam Induced Processing (FEBIP) for Emerging Applications in Carbon Nanoelectronics, Appl. Phys. A – Mat. Sci. & Proc. 117 (2014) 1659-1674.

[5] J. D. Fowlkes, R. Winkler, B. B. Lewis, M. G. Stanford, H. Plank, P. D. Rack, Simulation-Guided 3D Nanomanufacturing via Focused Electron Beam Induced Deposition, ACS Nano. 10 (2016) 6163-6172.

[6] L. Skoric, D. Sanz-Hernández, F. Meng, C. Donnelly, S. Merino-Aceituno, A. Fernández-Pacheco, Layer-by-Layer Growth of Complex-Shaped Three-Dimensional Nanostructures with Focused Electron Beams, Nano Lett. 20 (2020) 184-191.

[7] J. D. Fowlkes, R. Winkler, B. B. Lewis, A. Fernández-Pacheco, L. Skoric, D. Sanz-Hernández, M. G. Stanford, E. Mutunga, P. D. Rack, H. Plank, High-Fidelity 3D-Nanoprinting via Focused Electron Beams: Computer-Aided Design (3BID), ACS Appl. Nano Mater. 1 (2018) 1028-1041.

[8] M. R. Henry, S. Kim, A. G. Fedorov, High Purity Tungsten Nanostructures via Focused Electron Beam Induced Deposition with Carrier Gas Assisted Supersonic Jet Delivery of Organometallic Precursors, J. Phys. Chem. C 120 (2016) 10584−10590.

[9] M. M. Shawrav, P. Taus, H. D. Wanzenboeck, M. Schinnerl, M. Stöger-Pollach, S. Schwarz, A. Steiger-Thirsfeld, E. Bertagnolli, Highly Conductive and Pure Gold Nanostructures Grown by Electron Beam Induced Deposition, Sci. Rep. 6 (2016) 34003.

[10] M. R. Henry, S. Kim, K. Rykaczewski, A. G. Fedorov, Inert Gas Jets for Growth Control in Electron Beam Induced Deposition, APL. 98 (2011) 263109.

[11] S. Kim, M. R. Henry, Y. H. Moon, A. G. Fedorov, Multimode Jetting Unlocks a Trade-Off between Nanostructure Morphology and Composition in Focused Electron Beam Induced Deposition, Mater. Today Commun. (2019) 100645.

[12] K. T. Kohlmann, M. Thiemann, W. H. Brünger, -Beam Induced X-Ray Mask Repair with Optimized Gas Nozzle Geometry, Microelectron. Engr. 13 (1991) 279-282.

[13] I. Utke, V. Friedli, S. Amorosi, J. Michler, P. Hoffmann, Measurement and Simulation of Impinging Precursor Molecule Distribution in Focused Particle Beam Deposition/Etch Systems. Microelectron. Engr. 83 (2006) 1499-1502.

[14] S. Wachter, M. Gavagnin, H. D. Wanzenboeck, M. M. Shawrav, D. Belić, E. Bertagnolli, Nitrogen as a Carrier Gas for Regime Control in Focused Electron Beam Induced Deposition, Nanotechnol. 1 (2014) 16–22.

[15] M. R. Henry, S. Kim, A. G. Fedorov, Non-Equilibrium Adatom Thermal State Enables Rapid Additive Nanomanufacturing, Phys. Chem. Chem. Phys. 21 (2019) 10449-10456.

[16] M. Sabouri, M. Darbandi, Numerical Study of Species Separation in Rarefied Gas Mixture Flow through Micronozzles using DSMC, Phys. Fluids 31 (2019) 042004.

[17] G. A. Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Oxford University Press, New York, 1994.

[18] M. S. Ivanov, G. N. Markelovf, S. F. Gimelshein, Statistical Simulation of Reactive Rarefied Flows: Numerical Approach and Applications, AIAA 1998.

[19] J. S. Wu, K. C. Tseng, F. Y. Wu, Parallel Three-Dimensional DSMC Method Using Mesh Refinement and Variable Time-Step Scheme, Comput. Phys. Commun. 162 (2004) 166–187.

[20] K. C. Kannenberg, I. D. Boyd, Strategies for Efficient Particle Resolution in the Direct Simulation Monte Carlo Method, J. Comput. Phys. 157 (2000) 727–745.

[21] R. G. Wilmoth, A. B. Carlson, G. J. LeBeau, DSMC Grid Methodologies for Computing Low-Density, Hypersonic Flows about Reusable Launch Vehicles, AIAA 1812 (1996).

[22] M. Darbandi, E. Roohi, Applying a Hybrid DSMC/Navier–Stokes Frame to Explore the Effect of Splitter Catalyst Plates in Micro/Nanopropulsion Systems, Sensor Actuat. A-Phys. 189 (2013) 409– 419.

[23] A. L. Garcia, J. B. Bell, W. Y. Crutchfield, B. J. Alder, Adaptive Mesh and Algorithm Refinement Using Direct Simulation Monte Carlo, J. Comput. Phys. 154 (1999) 134–155.

[24] O. Aktas, N. R. Aluru, A Combined Continuum/DSMC Technique for Multiscale Analysis of Microfluidic Filters, J. Comput. Phys. 178 (2002) 342–372.

[25] J. S. Wu, Y. Y. Lian, G. Cheng, R.P. Koomullil, K.C. Tseng, Development and Verification of a Coupled DSMC–NS Scheme Using Unstructured Mesh, J. Comput. Phys. 219 (2006) 579–607.

[26] Y. Y. Lian, Y. S. Chen, K. C. Tseng, J. S. Wu, B. Wu, L. Yang, Improved Parallelized Hybrid DSMC–NS Method, Comput. Fluids 45 (2011) 254–260.

[27] Z. Li, Z. Li, H. Li, Y. Yang, X. Jiang, N-S/DSMC Hybrid Simulation of Hypersonic Flow over Blunt Body Including Wakes. AIP Conf. Proc. 1628 (2014) 519-526.

[28] E. Meiburg, Comparison of the Molecular Dynamics Method and the Direct Simulation Monte Carlo Technique for Flows around Simple Geometries, The Physics of Fluids 29 (1986) 3107.

[29] W. W. Liou, Y. Fang, Microfluid Mechanics – Principles and Modeling, McGraw-Hill Nanoscience and Technology Series, 2006.

[30] O. Tumuklu, D.A. Levin, J. M. Austin, Shock-Shock Interactions for a Double Wedge Configuration in Different Gases. in: Proceedings of the 53rd AIAA Aerospace Sciences Meeting, AIAA, FL, January 2015.

[31] M. N. Macrossan, A Fast Simulation Method with Arbitrary Viscosity Law, in: Proceedings of the 24th International Symposium on Rarefied Gas Dynamics, Italy, 2004.

[32] G. A. Bird, Monte-Carlo Simulation in an Engineering Context, Prog. In Astro, and Aero. 74 (1981) 239-255.

[33] D. B. Hash, J. N. Moss, H. A. Hassan, Direct Simulation of Diatomic Gases Using the Generalized Hard Sphere Model, J. Thermophys. Heat Tr. 8 (1994) 758-764.

[34] K. Koura, H. Matsumoto, Variable Soft Sphere Molecular Model for Inverse-Power-Law or Lennard-Jones Potential, Phys. Fluids A-Fluid. 3 (1991) 2459-2465.

[35] M. A. Gallis, J. R. Torczynski, D. J. Rader, G. A. Bird, Convergence Behavior of a New DSMC Algorithm, J. Comput.

Phys. 228 (2009) 4532–4548.

[36] J. N. Moss, DSMC Computations for Regions of Shock/Shock and Shock/Boundary Layer Interaction, AIAA 1027 (2001).

[37] K. Nanbu, Y. Watanabe, S. Igarashi, Conservation of Angular Momentum in the Direct Simulation Monte Carlo Method, J. Phys. Soc. Jpn. 57 (1988) 2877-2880.

[38] G. A. Bird. Forty Years of DSMC, and Now? AIP Conf. Proc. 585 (2001) 372-380.

[39] S. J. Robertson, D. R. Willis, Method-of-Characteristics Solution of Rarefied, Monatomic Gaseous Jet Expansion into a Vacuum, AIAA J. 9 (1971) 291-296.

[40] M. R. Henry, S. Kim, K. Rykaczewski, A. G. Fedorov, Inert Gas Jets for Growth Control in Electron Beam Induced Deposition, Appl. Phys. Lett. 98 (2011) 263109.

[41] S. Kim, M. R. Henry, A. G. Fedorov, Using an Energized Oxygen Micro-Jet for Improved Graphene Etching by Focused Electron Beam, Appl. Phys. Lett. 107 (2015) 233102.

[42] K. Kolasinski, Surface Science – Foundations of Catalysis and Nanoscience, John Wiley & Sons, 2012.

[43] K. Rykaczewski, W. B. White, A. G. Fedorov, Analysis of Electron Beam Induced Deposition (EBID) of Residual Hydrocarbons in Electron Microscopy, J. Appl. Phys. 101 (2007) 054307.