

Clustered Ensemble Averaging: A Technique for Visualizing Qualitative Features of Stochastic Simulations

Kayne M. Smith, David C. Banks*, Neil Druckman, Kevin Beason, and M. Yousuff Hussaini

School of Computational Science, Florida State University, Tallahassee, FL 32306, USA

Stochastic simulations of fluid flow deal with the evolution of uncertainties in initial and boundary conditions, parameters, and physical models. These uncertainties may lead to qualitatively different behaviors of the flow, raising the question of how to visualize random flow variables in a meaningful way. Our approach is to cluster the simulation data by qualitative phenomenon and then to compute average flow quantities within each cluster, a technique we call “clustered ensemble averaging.” This tactic extends the basic visualization strategy of extracting features from within a computational domain defined within time and space. The abstract features we seek are defined across the space of all possible simulations: each feature (cluster) is the subspace of simulations that share a basic behavior. We illustrate this technique on data from molecular dynamics simulations of laser-assisted particle removal, where the explosive evaporation of a laser-heated fluid acts on minute particle contaminants on a substrate, sometimes removing them and sometimes failing to do so. We ran the simulation a hundred times, comprising a hundred samples of the simulation space, and clustered each according to the behavior it exhibits. The images that result from clustered ensemble averaging reveal characteristics common to each cluster that are hidden when the average is calculated over all outcomes.

Keywords: Molecular Dynamics Simulations, Stochastic Processes, Uncertainty Visualization, Laser-Assisted Particle Removal.

1. INTRODUCTION

One of the main strategies of visualization is to display salient features in a dataset so they can be interpreted by the scientist, engineer, or physician who is analyzing the data. For three-dimensional (3D) scalar fields, feature-extraction techniques have been developed to display boundaries of tissue in medical data.^{1,2} For 3D vector fields, feature-extraction techniques have been developed to display shock fronts³ and vortex cores.^{4–6} Some of these techniques have been extended to extract features from 4D time-varying datasets, where the feature extends across both space and time.^{7–10} Additional techniques capture features that exist in 4D scale space, where cross-sections at different values of the scale axis correspond to different resolutions of the data.^{11–13} These examples illustrate feature-detection techniques that have been developed to locate and display features in two or three spatial dimensions, and in multidimensional domains of space-time or scale

space. Other visualization techniques merely approximate the geometric appearance of a feature, matching its shape to a best-fit model such as an ellipsoid.¹⁴ This approach can be understood as finding an average approximating shape among all features in a certain equivalence class.

We extend the idea of visualization via feature-detection to a new level of abstraction by considering a feature that exists across not only in time and space, but also across a set of variations. These variations arise from multiple runs of a stochastic (random) simulation starting from different initial conditions.

The three-body problem is a simple example that illustrates how diverse outcomes can result from minute variations of initial conditions: in the plane, three particles of equal mass are assigned positions $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3$ and velocities $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ at time $t = 0$. Each particle has four independent parameters (two for position and two for velocity), so the planar three-body problem is characterized by a 12-dimensional parameter space. Although most solutions of the planar three-body problem appear random and are of minimal interest, occasionally a periodic solution is obtained.

*Author to whom correspondence should be addressed.

Finding parameters that produce the qualitative behavior of periodicity is still an area of active research.¹⁵ In comparison, a computational molecular dynamics simulation of a thin layer of fluid may have more than a thousand particles. Although the 4000-dimensional parameter space of initial conditions is very large, the number of qualitatively distinct outcomes may be small.

Stochastic processes may be intuitively defined as systems which evolve probabilistically in time. Mathematically described, a stochastic process is a time-dependent random variable or field that evolves in time. In the present context, these processes are obtained as solutions to evolution equations with random initial and boundary conditions, and/or random physical and geometric parameters. The stochastic simulations of a flow configuration (created by some set of external conditions) are similar in some sense but differ from one another in detail. In order to analyze the underlying phenomena in such flows, one needs to define the mean value of the flow properties. A meaningful mean value is defined by the statistical average of the ensemble of all similar flows. This permits decomposition of a field variable into a mean and a fluctuation (deviation). The mean field then allows one to extract the “deterministic” organized structures underlying the random flow. The visualization of probabilistic evolution of fluid flows can be considered a form of “uncertainty visualization.”^{16, 17}

We present a technique for displaying the mean behaviors of a flow that exhibits two qualitatively distinct outcomes. Multiple runs of the simulation yield datasets that are assigned to one of two clusters. Ensemble averaging within one cluster reveals behavior that is not evident from taking an overall average. We demonstrate the technique on an example: a simulation of laser-assisted particle removal.

The remainder of the paper is organized as follows. Section 2 provides background into laser-assisted particle removal. Section 3 describes the computational simulation model. Section 4 describes how the particle data from the simulation is converted into continuous density fields and how these datasets are clustered. Section 5 describes how the particle data from the simulation is converted into continuous velocity data. Section 6 summarizes the results.

2. LASER-ASSISTED PARTICLE REMOVAL

Manufacturing processes for hi-tech devices in microelectronics or data storage continue to demand higher resolution, smaller linewidths, miniaturization, and vanishing mechanical clearances. Since it is not currently possible to totally eliminate all contamination sources during industrial processing, there is an ever-increasing need to develop methods for removing minute particle contamination from critical surfaces. Currently, the most widely used industrial cleaning techniques use liquid chemicals, which are themselves sources of contamination. Removing these particle contaminants is difficult since a submicron particle can

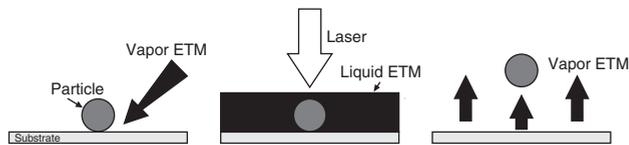


Fig. 1. Laser assisted particle removal (LAPR). Left panel shows circular particle on substrate surface. Vaporized energy transfer medium (ETM) is directed toward the substrate surface. The ETM condenses on the surface as shown in the middle panel. Laser energy is then directed toward the particle and liquid ETM, causing explosive boiling of the ETM and removal of the particle, as shown in the right panel.

adhere to a surface with a force that is over a million times its weight.¹⁸ A cleaning technique for submicron particles should be effective for particle removal, not cause surface damage, or add any extra contamination to the surface. Additionally, we want the process to be efficient, simple, fast, and chlorofluorocarbon-free.

Laser assisted particle removal (LAPR) is a technique, which meets the criteria listed above, for removing small particles (as small as 50 nm, though currently used principally for particles $\sim 1 \mu\text{m}$ in diameter) from a substrate. In this process, a small amount of energy transfer medium (ETM), usually a mixture of water and alcohol, is condensed onto the substrate, and then laser energy is directed at the substrate surface, as shown in Figure 1. This energy is absorbed by the ETM fluid, and perhaps also by the substrate and/or the particle, and results in removal of the ETM and sometimes in the removal of the particle. Particle removal is thought to be caused by a combination of three mechanisms: explosive boiling of the liquid ETM, thermal expansion of the particle (“hopping” effect), and thermal expansion of the substrate (“trampoline” effect).¹⁹

A single optimized laser pulse has been shown to effectively remove 90% of particles with sizes on the order of $1 \mu\text{m}$.²⁰ Repeating the process allows removal efficiency of nearly 100% to be achieved, typically within five to eight repetitions.²⁰ Two major factors influence the particle removal efficiency of LAPR:

- (1) the adhesion forces holding the particle to the substrate surface, and
- (2) the laser-induced particle removal forces.

It is well known that the cleaning efficiency increases with increasing laser fluence, but at very high laser fluences substrate surfaces are easily damaged by laser irradiation. Thus, determining the optimal laser cleaning conditions and clearly understanding the interaction mechanisms between particle and substrate surface are the goals of modeling the LAPR process.

3. SIMULATION MODEL

A two-dimensional molecular model was constructed for simulating LAPR. The model consists of a substrate, a particle, and the ETM fluid. To reduce computational requirements, a very simple molecular model was applied. All interactions between molecules in the ETM fluid,

Table I. Lennard-Jones potential parameters for interactions between the ETM, particle, and substrate.

| Interaction | σ | ε |
|--------------------|----------|---------------|
| ETM–ETM | 1.0 | 1.0 |
| ETM–particle | 1.0 | 1.5 |
| Particle–particle | 1.0 | 10.0 |
| ETM–substrate | 1.0 | 1.5 |
| Particle–substrate | 1.0 | 1.5 |

substrate, and particle were modeled using the Lennard-Jones 12-6 potential:²¹

$$\phi(r_{ij}) = 4\varepsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right] \quad (1)$$

where r_{ij} is the distance between molecules i and j , σ is a measure of the molecule's diameter (the distance where the energy is zero), and $-\varepsilon$ is the minimum value of the energy. The simulations were run using reduced (nondimensional) units,²¹ with the mass of all molecules equal. (For convenience, in addition to reduced units, we will present dimensional units assuming argon molecules; for argon, $\sigma = 0.43$ nm and $\varepsilon = 1.67 \times 10^{-21}$ J.) The Lennard-Jones potential parameters for the interactions between the ETM fluid, particle, and substrate are given in Table I.

The substrate is composed of five layers of molecules arranged in a hexagonal lattice, which is the two-dimensional crystal structure of the Lennard-Jones solid. During the simulation, the molecules of the substrate interact (via the potential function) with nearby fluid and particle molecules, but the substrate molecules are held in fixed positions. This is partly for additional computational cost savings, but mainly because thermal expansion of the substrate in the periodic geometry would likely cause buckling, which would substantially complicate our analysis. The rigid substrate approximation avoids this, although it does mean that the “trampoline” mechanism for particle removal is not present in these simulations.

The particle is circular, with a diameter of 19σ (6.46 nm), and is also composed of molecules in a hexagonal arrangement. The lowest layer of molecules that compose the particle is in contact with the molecules in the uppermost layer of the substrate.

Simulations were conducted on a two-dimensional computational domain of size $x = 202\sigma$ (68.68 nm) by $y = 240\sigma$ (81.6 nm). The domain is periodic in x and molecules that reach the maximum y -value are reflected back into the domain. This reflection is necessary to prevent the entire fluid from evaporating through the top of the domain. Since the important part of the simulation occurs in the lower half of the domain, the reflection condition has no effect on the results that follow.

After placing a particle on the substrate within a computational cell, fluid is added to the cell by performing a Grand Canonical Monte Carlo (GCMC) simulation. GCMC is a stochastic technique in which the number

of molecules is allowed to vary according to a specified chemical potential.²¹ This is accomplished through “moves” in which molecules are either displaced, created at random positions, or destroyed. When completed, as determined by stabilization of the total number of molecules, a liquid (ETM) film of varying thickness overlies the substrate and the particle, as shown in the first column of Figure 2.

Because particle removal is a discrete event, in order to determine removal efficiency it is necessary to simulate the removal process many times, starting from various initial conditions. Additional configurations were generated by allowing the GCMC simulation to run for an additional short time (100 cycles). Initial configurations for an ETM thickness of 50σ (17 nm) are shown in the leftmost column of Figure 2.

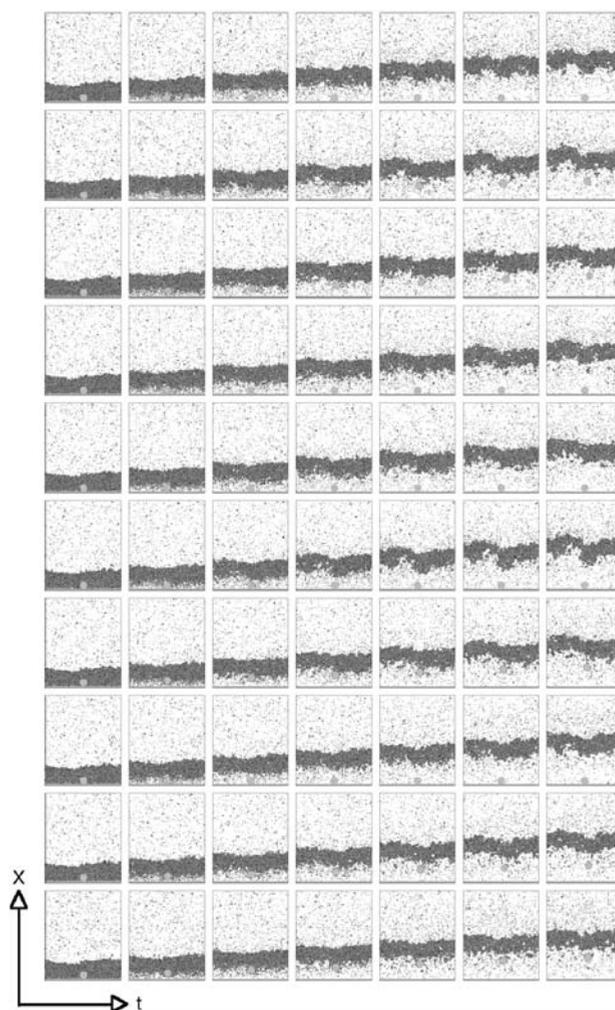


Fig. 2. Snapshots from LAPR simulations. The substrate is shown as a dark gray band at the bottom of each panel, the ETM fluid molecules are black, and the particle, which starts in contact with the substrate, is circular and gray. Each row represents a simulation that began with slightly different initial conditions. Each column is time advanced by 4,000 iterations (0.044 ns). Substrate temperature is 1.7 (205.7 °K). ETM fluid thickness is 50σ (17 nm). In the right column, we see that particle ejection occurs in half of the simulations.

In the molecular dynamics simulations, a fifth-order Gear predictor-corrector algorithm was used to integrate the equations of motion, with a time step of 0.005 (5.5×10^{-17} s). After 250 time steps, the substrate temperature was instantaneously increased from its base value of 0.4 (48.4 °K) to a value between 0.5 (60.5 °K) and 5.0 (605 °K). Increasing the substrate temperature in this manner approximates the effect of laser heating in the case where the laser energy is absorbed by the substrate, rather than by the fluid. For more details on the LAPR simulations, see Ref. [22].

Each simulation was run for 30,000 time steps (representing 0.33 ns of physical time), which, except in the case of temperature equal to 0.5 (60.5 °K), was enough to observe the complete ejection of the ETM fluid layer, and, in some cases, particle removal. Typical results are shown in Figure 2. In this figure, each column is time advanced by 4,000 iterations (0.044 ns), and each row began with slightly different initial conditions (i.e., number of ETM molecules, position of ETM molecules, and velocity of ETM molecules). The substrate temperature for each of these simulations was 1.7 (205.7 °K). By examining the rightmost column, we see that of the ten simulations, five resulted in particle removal, and five did not. Since the initial conditions for the set of ten simulations varied only slightly, and the substrate temperature was identical for each simulation, why was the particle removed in some simulations and not in others?

4. DATASET VISUALIZATION

Before describing the process of visualizing the fluid flow to answer this question of why the particle is sometimes removed, we begin by defining variables. Let T be the substrate temperature, t be the elapsed time, W be the width of the ETM fluid layer, and X represent the simulation number (or the initial conditions of the simulation). Then, Figure 2 shows t increasing to the right and X increasing upward, while $T = 1.7$, and $W = 50\sigma$.

We can also fix the elapsed time, t , and the temperature, T , and vary the initial conditions, X , and the ETM thickness, W , as shown in Figure 3. This figure consists of snapshots from LAPR simulations at $t = 30,000$ (0.33 ns), and $T = 1.7$ (205.7 °K), where in this case, the ETM fluid thickness, W , varies from very thin (6σ or 2.04 nm), to very thick (70σ or 23.8 nm). Again, notice the range of behaviors that occur in LAPR: sometimes the particle lifts with the ETM fluid, and sometimes it doesn't, even as the thickness of the fluid layer varies. Looking at each snapshot in this manner, it is difficult to extract meaningful data from the simulations since the dimension is so large.

We get a better picture of the average response for a given T and W by averaging over different initial conditions; that is, we integrate out the variable X . This gives rise to Figure 4, which shows fluid thickness, W , increasing to the right, and temperature, T , increasing upward

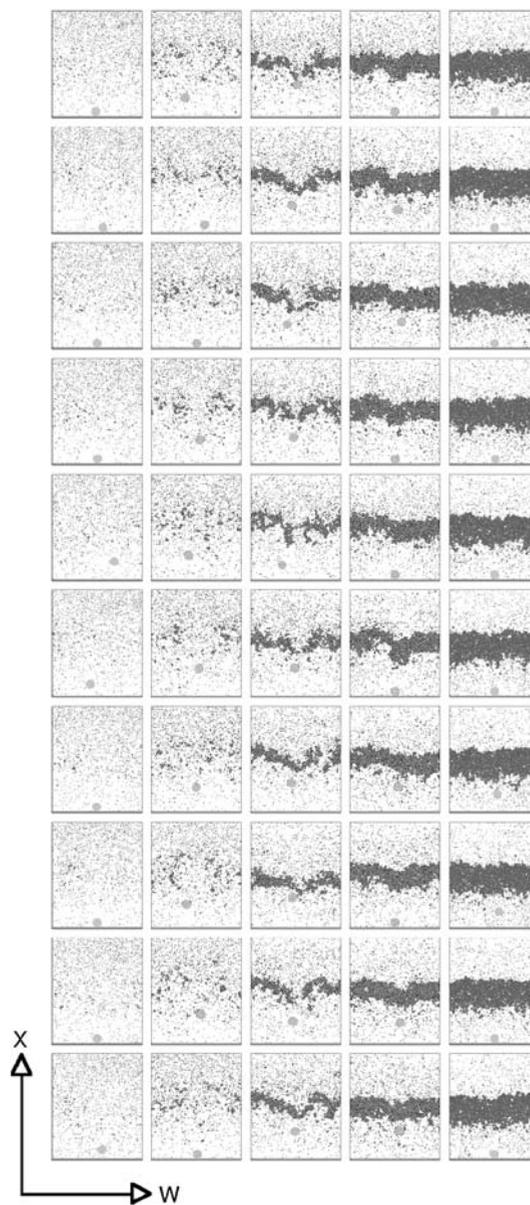


Fig. 3. Snapshots of LAPR simulations. For all simulations, the substrate temperature, T , is 1.7 (205.7 °K) and the elapsed time, t , is 30,000 (0.33 ns). Each column shows a different ETM thickness, W , with W increasing to the right, and taking on the values 6σ (2.04 nm), 10σ (3.4 nm), 25σ (8.5 nm), 50σ (17 nm), and 70σ (23.8 nm). Each row shows a simulation that began with slightly different initial conditions, X .

from 1.5 (181.5 °K) to 2.0 (242 °K), by 0.1 (12.1 °K). In particular, the third row from the bottom shows the results of averaging out all the X 's from Figure 3. Notice there is a bimodal appearance of the particle, especially in the middle columns, where you can see an average position of the particle in contact with the substrate and another average position between the substrate and the average lifted fluid layer. Each of these two modes corresponds to a subset (called an ensemble) of the simulations shown in Figure 3. Rather than averaging all of the simulations, we can instead average only those simulations in the first ensemble, where the particle is not removed, as shown in

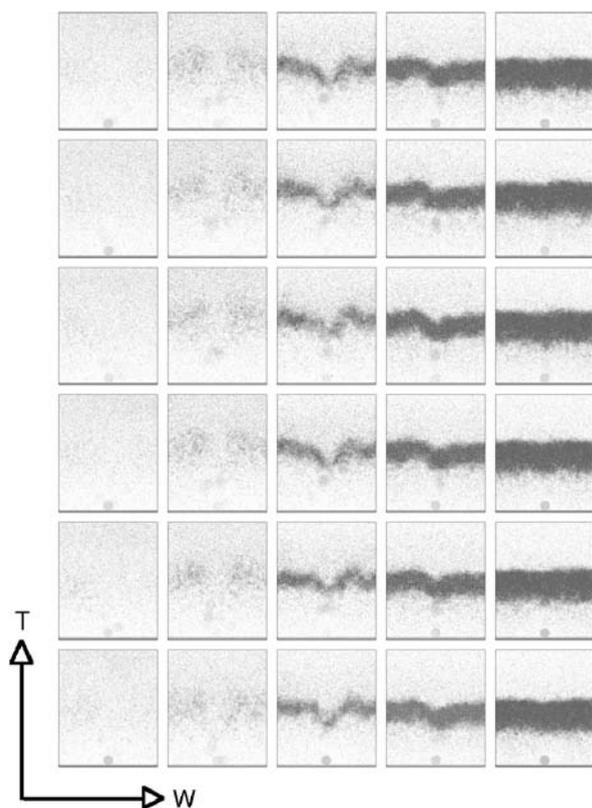


Fig. 4. Averages of snapshots of LAPR simulations. Elapsed time t is 30,000 (0.33 ns). ETM thickness W increases to the right and substrate temperature T increases upward, from 1.5 (181.5 °K) to 2.0 (242 °K) by 0.1 (12.1 °K). Note that the combination of high temperature and medium width (upper row, middle column) results, on average, in particle removal.

the upper panel of Figure 5, and then average the simulations in the second ensemble, where the particle is removed, as shown in the lower panel of Figure 5. The missing entries (empty rectangles) indicate that there is no data. For example, at the bottom right of the lift ensemble (lower panel) we see that particle removal did not occur in any of the simulations.

For the simulations presented thus far, there were 10 initial conditions, X , for each ETM thickness, W , and temperature, T . For comparison, we also conducted 90 additional simulations for the case $T = 1.6$ (193.6 °K), and $W = 50\sigma$ (17 nm), making a total of 100 simulations for that case. Figure 6 shows the average of the initial 10 simulations (left panel) and the average of all 100 simulations (right panel) at $t = 30,000$ (0.33 ns). With more simulations to average, the variance is reduced and the resulting average image looks much smoother, but at the expense of more compute time. We note that each simulation took about 14 minutes on a desktop computer with a 1.2 GHz processor and 256 MB of RAM.

Although 100 simulations provides a smoother average, the trajectory of the particle is difficult to perceive. By calculating the ensemble averages of only the particle molecules (averaged over both time, t , and initial

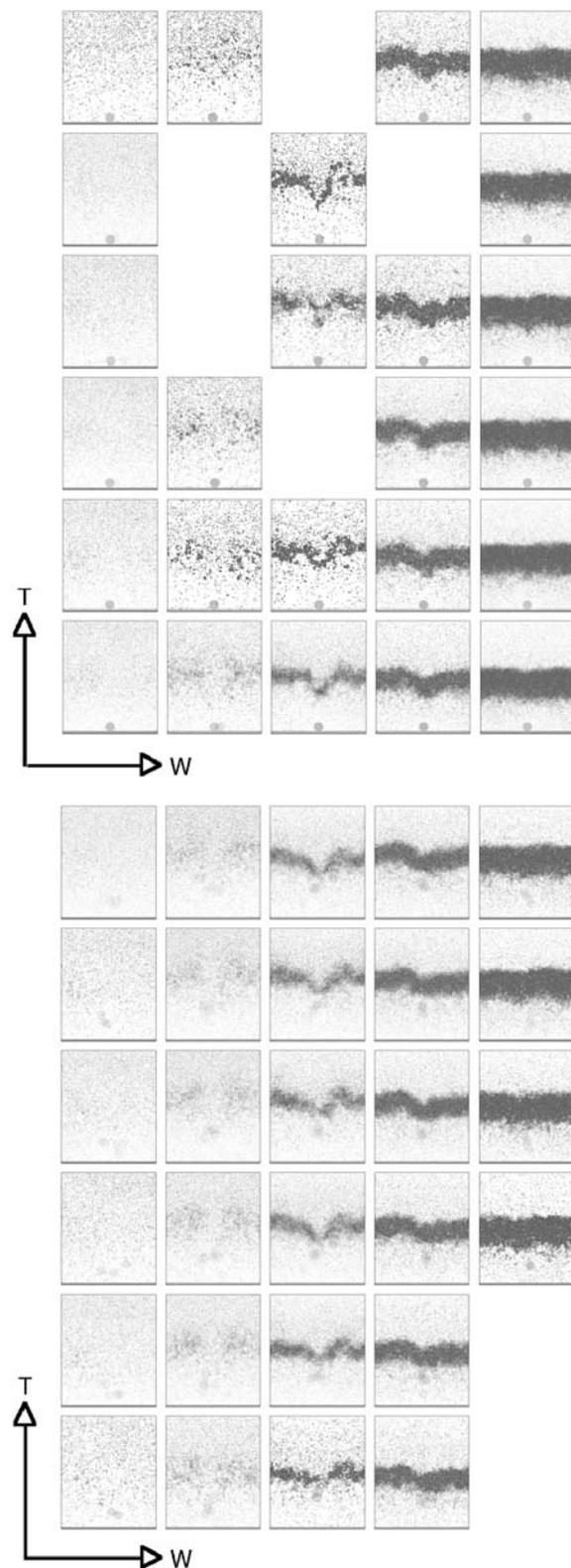


Fig. 5. Ensemble averages of LAPR simulation snapshots for simulations where there is not particle removal (upper) and for simulations where there is particle removal (lower). Elapsed time t is 30,000 (0.33 ns). ETM thickness W increases to the right, and substrate temperature T increases upward, from 1.5 (181.5 °K) to 2.0 (242 °K).

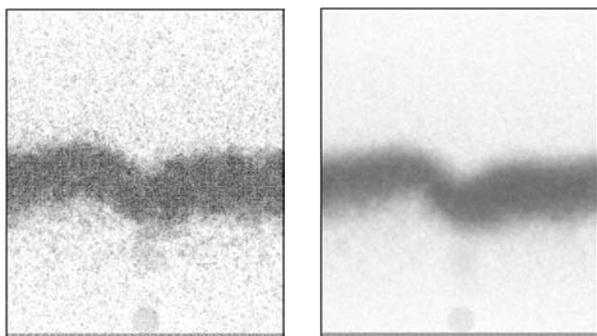


Fig. 6. Averages of 10 (left) and 100 (right) simulations at temperature $T = 1.6$ (193.6 °K), elapsed time $t = 30,000$ (0.33 ns), and ETM thickness $W = 50\sigma$ (17 nm).

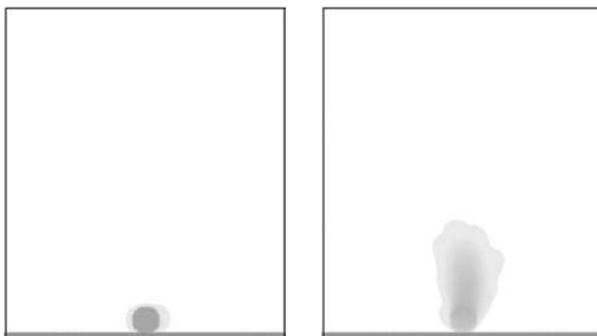


Fig. 7. Ensemble averages (over time t and simulation X) of particle trajectories for the case of no particle removal (left) and particle removal (right). $T = 1.6$ (193.6 °K), and $W = 50\sigma$ (17 nm). (The contrast was adjusted to enhance the image.)

conditions, X), the average particle trajectories are easily seen, as in Figure 7. On the left, is the ensemble average trajectory of the particle for the 30 simulations where particle removal did not occur. In this case, it can be seen that sometimes the particle rolls to the left or right, but the most likely (darkest) case is no motion at all. On the right is the ensemble average trajectory of the particle for the 70 simulations where particle removal did occur. In this case, it can be seen that the most likely (darkest) case is that the particle lifts off the substrate perpendicular to the substrate, although it is also possible that the particle will lift-off the substrate at a slightly different angle.

5. SMOOTHED DENSITY IMAGES

A faster way to produce a smoothed image of the molecular density (as shown in Fig. 6) is illustrated in the upper panels of Figure 8. The upper left panel shows the molecules in space with a grid laid over the domain. Each molecule is assigned a density function

$$f_i(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{p}_i) \tag{2}$$

where \mathbf{p}_i is the position of i and δ is the Dirac delta distribution. We convolve $f = \sum f_i$ with a filter function $h(x)$

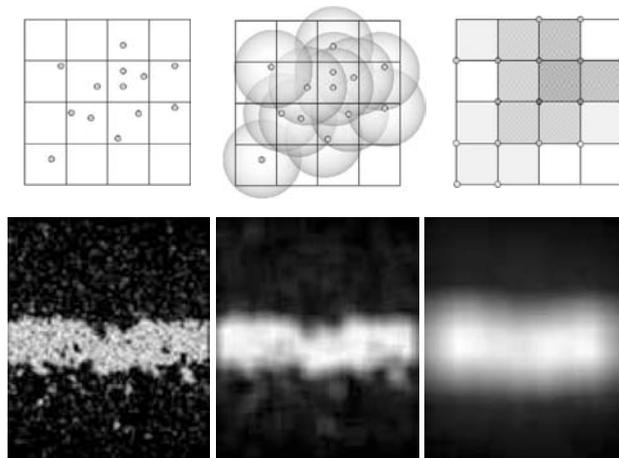


Fig. 8. Smoothed density images. Upper panels illustrate how smoothed density images are computed: left panel shows molecules in space with an overlaid grid, middle panel shows molecules each with its density function centered at the molecule’s position, and right panel shows the resulting density image. Lower images are examples of smoothed density images with varying filter sizes. The filter radii used, from left to right, are 1σ (0.34 nm), 8σ (2.72 nm), and 32σ (10.88 nm).

to blur the molecular densities across the image, as shown in the upper middle panel of Figure 8. We choose a filter function with compact support, namely the cubic approximation to the Gaussian given by $h(x) = 2x^3 - 3x^2 + 1$, where x is scaled so that $x = 1$ at the filter radius, which is given in spatial coordinates. We can think of the resulting image, $I(\mathbf{x})$, as a scalar function from $\mathbb{R}^2 \rightarrow \mathbb{R}$, corresponding to the molecular density at a point in space, where I is given by

$$I(\mathbf{x}) = \frac{\sum_i f(\mathbf{x})h(\mathbf{x} - \mathbf{p}_i)}{\sum_i h(\mathbf{x} - \mathbf{p}_i)} \tag{3}$$

and is illustrated in the upper right panel of Figure 8. We normalize (divide by the sum of h) because the superposition of filter functions may sum to something other than one.

The lower panel of Figure 8 shows the result of this convolution on a given simulation, producing a sequence of images with different filter radii. The radii used, from

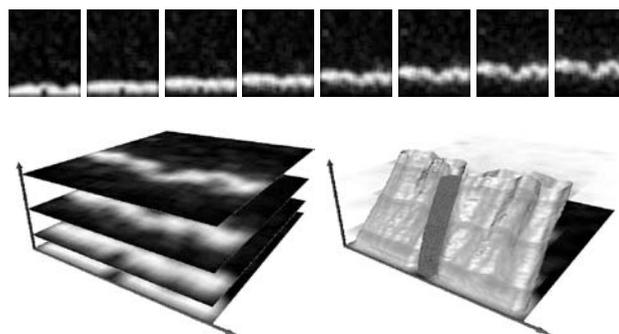


Fig. 9. Smoothed density images are used to make an isosurface. Upper panel: time sequence of smoothed density images. Lower left panel: Smoothed density slices stacked in time. Lower right panel: Isosurface of density, with particle trajectory shown as dark cylinder.

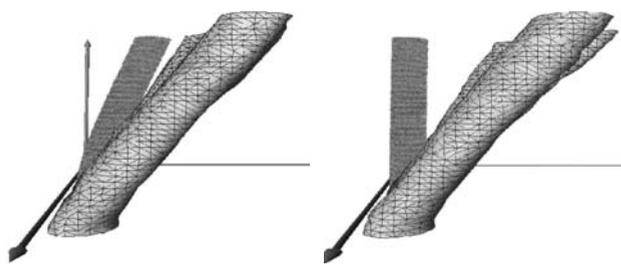


Fig. 10. Left: Close-up of density isosurface (right panel of previous figure) for a simulation where particle removal occurs. Right: Close-up of density isosurface for a simulation where particle removal does not occur. Time, t , increases in the vertical direction.

left to right, are 1σ (0.34 nm), 8σ (2.72 nm), and 32σ (10.88 nm). In each image the color black corresponds to low density (0 molecules per unit area), while white corresponds to high density (typically about 60 molecules per unit area). Thus, we have transformed a collection of about 8,000 molecules into a continuous scalar-valued density function.

The upper panel of Figure 9 shows smoothed density images for a time sequence of simulations where $T = 1.5$ (181.5 °K), and $W = 25\sigma$ (8.5 nm). If we treat the time direction as a third spatial coordinate, these density functions can be stacked to produce a 3D scalar field $I(x, y, t)$ as shown in the lower left panel of Figure 9. An isosurface of this scalar field with density value of about 30 is shown in the lower right panel of Figure 9, where we see the boundary of the subvolume which contains the bulk of the fluid molecules. This figure also shows the trajectory of the particle (as a dark cylinder). The bottom of the trajectory corresponds to the beginning of the simulation where the particle is on the substrate. At the top of the trajectory, where $t = 20,000$ (0.22 ns), the particle has moved away from the bottom of the domain, having been lifted by the ETM fluid which has moved past the particle.

Figure 10 shows the two lifting behaviors more clearly. With increasing time, the isosurface containing the volume of fluid moves away from $y = 0$. As in Figure 9, the time direction is aligned vertically, and the y direction of the image is aligned in the horizontal right direction. The left panel shows a run where the particle lifts and is removed, while the right panel shows a run where the particle does not lift. In both panels, notice that the ETM fluid closest to the substrate (on the left in the figures) begins to move before the fluid farthest from the substrate (on the right in the figures). This difference in velocities is visible as a kink in the isosurface near $t = 0$ on the right side of both figures.

6. VECTOR FIELDS AND ENSEMBLES

The previous section showed ensemble averages that distinguish qualitative behavior of the particle. Figures 7, 8, and 10 explicitly show examples of particle trajectories in the two regimes (particle removal and no-removal), and

of the ETM layer's trajectories in these two regimes. It is clear that two distinct behaviors occur; the question is, what property of the ETM causes the difference? The ETM always lifts whether the particle does or not, and at the crucial early stage of the simulation, the images of the ETM density are practically indistinguishable. We were therefore motivated to look at clustered ensemble averages of the ETM *velocity* for evidence of any subtle difference between the two regimes. This required us to produce a vector field from discrete particles in motion.

Just as we convolved molecular densities with a filter function to produce a continuous scalar density field, we can also convolve the molecular velocities with a filter function to produce a continuous velocity field. The upper panel of Figure 11 illustrates this process. The upper left panel shows several molecules each with a given velocity vector. A grid is laid over the domain. The upper middle panel of the figure shows the filter function centered at each of the molecular positions \mathbf{p}_i . The vector valued velocity at each grid location is found by summing the product of the filter function and the velocity for all molecules:

$$\mathbf{V}(\mathbf{x}) = \frac{\sum_i \mathbf{v}_i h(\mathbf{x} - \mathbf{p}_i)}{\sum_i h(\mathbf{x} - \mathbf{p}_i)} \quad (4)$$

Again, we normalize by dividing by the sum of h since the sum of filter functions applied at a grid point may be greater than one. As before, we used the cubic interpolating polynomial $h(x) = 2x^3 - 3x^2 + 1$, for the filter function, where x is scaled so that $x = 1$ at the filter radius, which

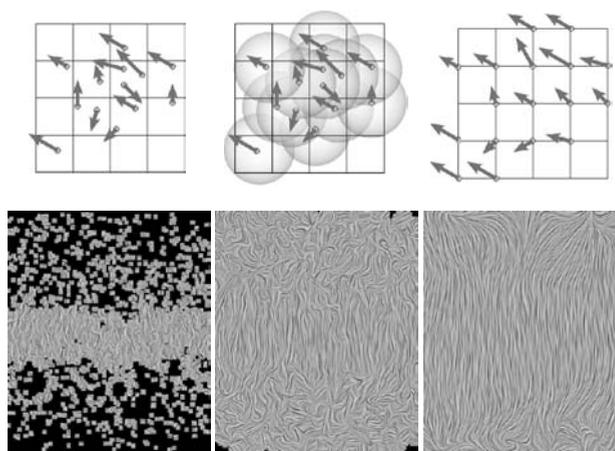


Fig. 11. Continuous velocity fields. Upper panels show how a continuous velocity field is constructed from molecular velocities. Upper left panel: molecules in space with a grid overlaid. Upper middle panel: each molecule is shown with the filter function centered at the molecules position. Upper right panel: after summing the product of the filter function with the velocity, and normalizing, the result is a continuous vector field on the grid. Lower panels show continuous vector fields using line integral convolution. Images of continuous vector fields are for the case of $T = 1.7$ (205.7 °K), $W = 50\sigma$ (17 nm), $t = 30,000$ (0.33 ns), and $X = 1$, with different filters of varying widths. The widths of the filters used, from left to right, are 1σ (0.34 nm), 8σ (2.72 nm), and 32σ (10.88 nm).

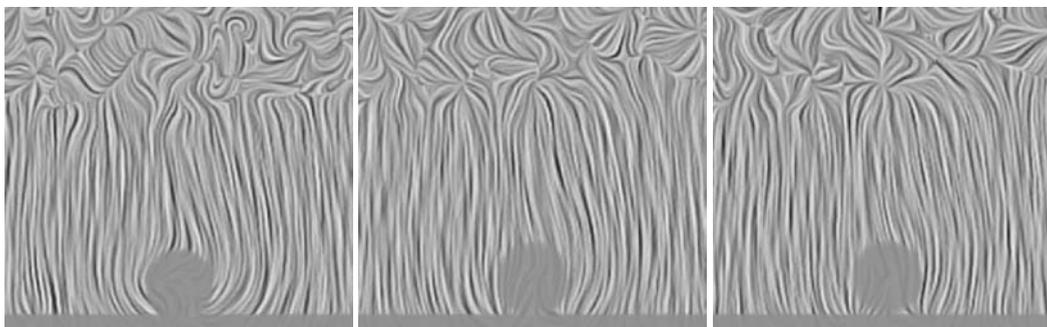


Fig. 12. Ensemble averages of molecular dynamics simulations of laser-assisted particle removal. Images are line integral convolutions of averaged fluid vector field data from the clustered set of simulations where particle removal does not occur (left), from the entire data set of 100 simulations (middle), and from the clustered set of simulations where particle removal does occur (right). The particle is circular and rests upon a substrate, both shown as dark gray. Note that in the left image, where particle removal does not occur, the flow field at the bottom of the particle is directed around the particle, but this behavior is not as pronounced in the middle image, where all data sets have been averaged. Visible in the right image, beneath the particle, is the vertically directed flow field that is acting to lift the particle.

is given in spatial coordinates. The final velocities on the grid are shown in the upper right panel of Figure 11.

The lower panels of Figure 11 show the continuous vector fields of a given simulation for three different filter radii using line integral convolutions (LIC)²³ to display the vector fields. The LIC images show streamlines of the flow. The solid black regions correspond to missing data where no molecule was close enough for its velocity to be blurred at that point. On the left, using a small radius of 1σ (0.34 nm), the sparseness of the data is apparent. The middle panel shows that by increasing the radius of the filter function to 8σ (2.72 nm), the missing data can be filled in, leaving missing data mostly at the bottom of the image. In the right panel, we have further increased the filter radius to 32σ (10.88 nm). We see that although it is possible to fill in the missing data making a dense vector field, much of this vector field corresponds to sparse data.

Although the images in the lower panels of Figure 11 were created using only one simulation, the same process can be applied to many simulations creating an image of the average velocity over many runs (namely, the ensemble-averaged velocity). We see the results of such an average in Figure 12. For the center panel of Figure 12, we averaged 100 simulations at $T = 1.6$ (193.6 °K), $t = 30,000$ (0.33 ns), and $W = 50\sigma$ (17 nm). For the left image, we averaged the ensemble of simulations for the cluster where particle removal did not occur (30 cases), and for the right image, we averaged the ensemble of simulations for the cluster where particle removal did occur (70 cases). In each of these images, we used a filter radius of 8σ (2.72 nm), which corresponds to the lower middle panel of Figure 11. In each of these images, the particle and substrate are shown as gray regions to allow these locations to be seen against the vector field. In the left panel (the cluster that does not lift) the velocity field at the base of the particle is clearly directed in an outward direction forming what looks like a wake around the particle, while in the middle panel (all images averaged together)

this behavior is not as pronounced. The center panel more closely matches the right panel where particle removal has occurred. In the right panel of Figure 12, in the space between the particle and the substrate, the velocity field beneath the particle can be seen acting on the particle in an upward direction, suggestive of the lifting force that successfully dislodges the particle from the substrate.

7. SUMMARY

By clustering the simulation data by qualitative phenomenon, and then computing average flow quantities within each cluster, we were able to enhance the flow quantities for each cluster that were hidden in the overall average flow quantities. Although we applied this technique to simulations of laser-assisted particle removal, clustering the particle removal and non-removal cases, this technique could be applied to other stochastic simulations of fluid flow.

In future work we want to address the question of optimal sample size determination, as well as investigate the use of the unsteady line integral convolution.

Acknowledgments: This research was supported in part by the National Science Foundation through grant ACI-9872140 to M. Y. Hussaini and ACI-0083898 to D. C. Banks.

References

1. T. F. Chan, J. Shen and L. Vese, Notices of American Mathematical Society (2003), Vol. 50.
2. Y. Sato, C.-F. Westin, A. Bhalerao, S. Nakajima, N. Shiraga, S. Tamura, and R. Kikinis, *IEEE Transactions on Visualization and Computer Graphics* 6 (2000).
3. K.-L. Ma, J. van Rosendale, and W. Vermeer, in Proceedings of the Symposium on Volume Visualization, IEEE Press, San Francisco (1996), pp. 87–95.
4. D. C. Banks and B. A. Singer, *IEEE Transactions on Visualization and Computer Graphics* 1 (1995).

5. D. N. Kenwright and R. Haines, in *IEEE Visualization* 413 (1997).
6. R. Peikert and M. Roth, in Proceedings of IEEE Visualization, edited by D. S. Ebert, M. Gross, and B. Hamann, IEEE Computer Society Press (1999), pp. 263–270.
7. P. Bhaniramka, R. Wenger, and R. Crawfis, in *Proceedings Visualization*, edited by T. Ertl, B. Hamann, and A. Varshney (2000), pp. 267–273.
8. D. Silver and X. Wang, *IEEE Transactions on Visualization and Computer Graphics* 3 (1997).
9. C. Weigle and D. C. Banks, in *IEEE Visualization*, edited by R. Yagel and G. M. Nielson (1996), p. 173.
10. T. Wischgoll, G. Scheuermann, and H. Hagen, in *Vision, Modelling, and Visualization*, Stuttgart, Germany (2001).
11. L. M. J. Florack, B. M. ter Haar Romeny, J. J. Koenderink, and M. A. Viergever, *J. Mathematical Imaging and Vision* 4 (1994).
12. T. Lindeberg, *Int. J. Comp. Vision* 30 (1998).
13. C. Weigle and D. C. Banks, in *Volume Visualization Symposium* 103 (1998).
14. F. Post, T. van Walsum, F. Post, and D. Silver, in *Visualization*, edited by G. Nielson and D. Silver, IEEE Computer Society Press, Los Alamitos (1995), pp. 288–295.
15. A. Chenciner and R. Montgomery, *Annals of Mathematics II. Series* 152 (2000).
16. S. Djurcilov, K. Kim, P. Lermusiaux, and A. Pang, *Computers and Graphics* 26 (2002).
17. D. Kao, A. Luo, J. L. Dungan, and A. Pang, in *Proceedings Information Visualization*, IEEE Computer Society (2002), pp. 219–225.
18. S. Kim and C. J. Lawrence, *Chem. Eng. Sci.* 43 (1998).
19. A. C. Tam, W. P. Leung, W. Zapka, and W. Ziemlich, *J. Appl. Phys.* 71 (1992).
20. S. D. Allen, A. S. Miller, and S. J. Lee, *Mater. Sci. Eng. B* 49 (1997).
21. M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids*, Oxford University Press, New York (2001).
22. K. M. Smith, M. Y. Hussaini, L. D. Gelb, and S. D. Allen, *Appl. Phys. A* 77 (2003).
23. B. Cabral and C. Leedom, in *Proceedings of ACM SIGGRAPH*, ACM (1993), pp. 263–270.

Received: 2 March 2006. Accepted: 14 March 2006.