Physics Extraction Techniques for High-Fidelity Atomization Simulations

C. Rubel, W. Krolick, M. Owkes*
Department of Mechanical and Industrial Engineering, Montana State University, USA
clark.rubel@student.montana.edu, william.krolick@msu.montana.edu and
mark.owkes@montana.edu

Abstract

Many research groups are capable of performing impressive high-fidelity simulations of atomizing jets that leverage advances to numerical methods and ever increasing computational resources. The simulations produce very large data-sets describing the flow and have the potential to advance our understanding of atomization. The challenge to making the results useful is extracting relevant physics from these large data-sets. In this work, we propose two physics extraction techniques that provide 1) the fundamental instabilities that exist on a jet's liquid core that dictate the largest structures generated during atomization and 2) the ancestry of droplets created as the coherent liquid core breaks into droplets and ligaments which may continue to break into smaller droplets. Understanding these processes will allow for low-fidelity atomization models to be developed and tested that agree with the physics predicted by detailed simulations.

Introduction

The liquid jet atomization process has many wide-ranging applications, fuel injection and cooling sprays being two examples. While the process is used frequently in everyday life, there is still much to be learned about it. Due to continued advances in computational power and numerical methods for gas-liquid flows, very accurate direct numeric simulations (DNS) have become feasible for many relevant flows.

Although direct numeric simulations have reached the point that they give high fidelity results and compare well to experimental data, they are too computationally demanding for many practical applications. Many research groups as well as industry use less computationally demanding (and less accurate) methods such as Reynolds Average Navier Stokes (RANS) and Large Eddy Simulations (LES) coupled with models to predict atomization physics such as droplet breakup. These models include, for example, Eulerian-Lagrangian Spray and Atomization (ELSA) model [1], Unified Spray Breakup (USB) model [2], Enhanced Taylor analogy breakup (TAB) [3].

Using DNS results to test and improve breakup models used in RANS or LES simulations will lead to more accurate predictions at lower costs. However, a significant challenge is the large amount of data produced by DNS, which can reach 100's of terabytes. Therefore, it is desirable to develop tools that can extract relevant physics from DNS results to 1) produce more accurate lower order models and 2) gain a better understanding of the physics of atomizing jets. The work presented here details techniques to extract two physical processes important for accurate atomization models: namely, the primary instability on the liquid core and the ancestry statistics of droplets.

The first physics extraction technique is to find a way to numerically categorize the instabilities on the coherent liquid core of an atomizing jet. The proposed method uses dynamic mode decomposition (DMD) of a representation of the jets liquid core from a DNS. Several studies have been done in attempt to categorize atomization jets [4, 5, 6, 7, 8], but the established utility of DMD is that it can more specifically categorize the different contributing dynamics that are driving the jet breakup. Studies such as [9, 10, 11] have been used to apply DMD directly to similar CFD applications. This work differs in that DMD is performed on positional data of a numeric jet rather than velocity, vorticity, or experimental results. Performing DMD on positional data could be significant in the creation of a lower order sub-grid atomizing model using information from DNS simulations. DMD is a type of reduced order modeling that takes a non-linear system (most complicated fluid systems) and evaluates it through time with a linear operator to extract spatial and temporal information from the system. If the instabilities that exist in a jets core can be numerically extracted, the results will be useful to understand the underlying physics that exist in an atomization jet. Other possible uses include finding relationships between the destabilization modes and standard input parameters like the Reynolds and Weber numbers or creating a type of machine learning algorithm that would not require the direct use of the computationally expensive equations that define fluid systems to represent the atomization.

The second physics extraction technique will compute droplet ancestry statistics by identifying when liquid structures break apart or merge together. The information, collected from DNS, will allow for statistics of droplet

*Corresponding author: mark.owkes@montana.edu

breakup dynamics to be produced and provide valuable insight on the development of breakup models. The extracted statistics will answer questions like "given a droplet of a given size and shape in a characterized flow, how long will the droplet last and how will the droplet break up?". The large number of droplet breakup and merge events that occur in an atomizing jet will each be identified, quantified, and stored in a graph database. Graph databases are designed to store large data sets connected through relationships (breakup and merge events). In this form the information on droplet ancestry can be readily queried allowing for details of atomization to be understood and breakup models to be tested and improved.

Numerical Methods

Both proposed physics extraction tools are implemented in the NGA computational platform [12] that solves the incompressible low Mach number, variable density Navier Stokes equations. The approach uses a staggered Cartesian mesh with a semi-implicit second-order Crank-Nicolson formulation similar to the method of Choi and Moin [13]. Away from the phase interface second-order finite difference operators that are well suited to simulate turbulent flows and conservatively transport mass, momentum, and any scalars [14]. Near the phase interface an unsplit geometric semi-Lagrangian volume-of-fluid (VoF) method is leveraged [15, 16]. The surface tension force is implemented using the ghost-fluid method [17] with the interface curvature computed with the ACES method [18].

Liquid Core Instability Extraction

In order to perform the dynamic mode decomposition to find the dominate instability on the liquid core, first a representation of the two-dimensional phase interface is needed. To do this, the jet's liquid core was defined as a two-dimensional matrix of radii $R(x, \theta)$ that are a function of the axial direction x and azimuthal angle θ, shown in Fig. 1. A code was written and integrated into the NGA framework [12] to compute R. The code marched along rays through the axial and θ directions, saving a radius at each ray/interface intersection. The rays shown in Fig.1 are defined by a line on the normal plane at the axial location and the angle θ relative to a constant reference vector in the same plane. The code only counted intersections of the ray and jet core interfaces, separate structures (droplets) were ignored. As perturbations on the jet's core grow, a single ray may intersect the interface multiple times due to the interface overlapping on itself. To handle this situation, multiple jet core intersection are averaged. This procedure provides $R(x, \theta)$, a two-dimensional representation of the jet's core geometry that is used as the input into the DMD analysis.

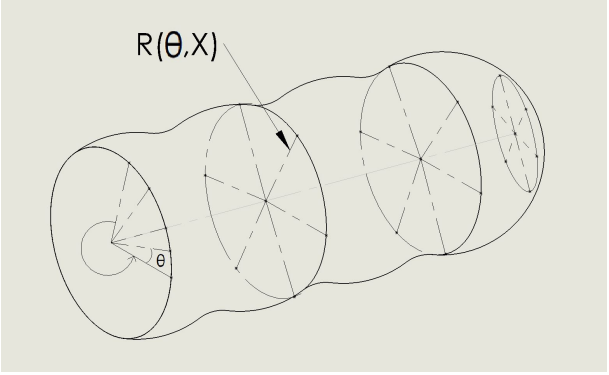

Figure 1. Jet core height $R(x, \theta)$ defined on a schematic of a liquid jet core.

The DMD analysis uses the jet's radius $R(x, \theta)$ and computes the dominate modes that exist in the spatially and temporally varying data. The method used was Exact DMD, presented by Kutz et al. [19] and shown in Algorithm 1. DMD uses the properties of a linear operator, computed with a least squares fit, to extract the individual low order modes that drive the system. Manipulations of the eigenvectors and eigenvalues of the linear operator represent the driving modes and how they vary through time, respectively. Using the properties of a psuedoinverse (step 3 in Algorithm 1) and singular value decomposition (SVD) the analysis establishes the least square fit to the system and solves for the operator. The SVD of the original data establishes the principal components of the system, then the DMD takes that information and describes what those principal components are doing through time.

Algorithm 1 Exact DMD Algorithm

1: Vectorize $R(x, \theta)$ from each time step and place into one large matrix $\boldsymbol{X}$, each column containing one time-step
2: Create two new matrices $\boldsymbol{X}_1$, $\boldsymbol{X}_2$ that span $\boldsymbol{X}(1 : end - 1)$ and $\boldsymbol{X}(2 : end)$, respectively
3: Set $\boldsymbol{X}_2 = \boldsymbol{A} * \boldsymbol{X}_1$ (therefore: $\boldsymbol{A} = \boldsymbol{X}_2 * \boldsymbol{X}_1^+$, where $\boldsymbol{X}_1^+$ is the Moore-Penrose Psuedoinverse of $\boldsymbol{X}_1$)
4: Take "economy" SVD of $\boldsymbol{X}_1$ so that $\boldsymbol{X}_1 = \boldsymbol{U} * \boldsymbol{S} * \boldsymbol{V}$'
5: Truncate SVD matrices for desired lower rank representation
6: Since $\boldsymbol{X}_2 = \boldsymbol{A} * \boldsymbol{X}_1$, then $\boldsymbol{X}_2 = \boldsymbol{A} * \boldsymbol{U} * \boldsymbol{S} * \boldsymbol{V}$'
7: $\hat{\boldsymbol{A}} = \boldsymbol{U}' * \boldsymbol{X}_2 * \boldsymbol{V} * \boldsymbol{S}^{-1}$ (from the projection of the full matrix $\boldsymbol{A}$ onto the principal components $\boldsymbol{U}$: $\hat{\boldsymbol{A}} = \boldsymbol{U}^* * \boldsymbol{A} * \boldsymbol{U}$)
8: Eigenvalue decomposition of $\hat{\boldsymbol{A}}$: $\hat{\boldsymbol{A}} * \boldsymbol{W} = \boldsymbol{W} * \boldsymbol{\lambda}$
9: Dynamic modes $\boldsymbol{\psi} = \boldsymbol{X}_2 * \boldsymbol{V} * \boldsymbol{S}^{-1} * \boldsymbol{W}$ (time dynamics represented by $\boldsymbol{\lambda}$)
10: Vector coefficients representative of initial conditions: $\boldsymbol{b}_o = \boldsymbol{\psi}^+ * \boldsymbol{X}_o$
11: Low rank recreation of system at each time step: $\hat{\boldsymbol{X}}_n = [\boldsymbol{\psi} * \boldsymbol{\lambda}^0 * \boldsymbol{b}_o, \boldsymbol{\psi} * \boldsymbol{\lambda}^1 * \boldsymbol{b}_o, \boldsymbol{\psi} * \boldsymbol{\lambda}^2 * \boldsymbol{b}_o, ..., \boldsymbol{\psi} * \boldsymbol{\lambda}^n * \boldsymbol{b}_o]$

Droplet Ancestry Extraction

The droplet ancestry extraction tool identifies splits and merges during a simulation and stores this information in an an ancestry database. The splits and merges are identified with two numbers. The first is the structure identification number $\mathcal{S}$, which is a unique number to each droplet or other liquid structure. The second is the liquid identification number $\mathcal{L}$, which moves with liquid parcels from one timestep to the next allowing for the ancestry of the droplets to be tracked. A merge occurs when multiple $\mathcal{L}$ are found within a single structure identified with a single $\mathcal{S}$. A split occurs when multiple $\mathcal{S}$ are found with the same $\mathcal{L}$. Details of this process is provided below.

The structure identifier $\mathcal{S}$ is a unique integer associated with each structure (droplet, ligament, etc.) within the flow domain. The structures are identified using the algorithm proposed by Herrmann [20]. This algorithm finds a computational cell that contains liquid starting a structure, then it adds neighboring cells that contain liquid until no additional liquid cells are found, completing the identification of this structure. The process is repeated until all unique structures are found, each with a unique identifier $\mathcal{S}$.

The liquid identifier $\mathcal{L}$ is a unique integer that moves with each liquid structure. The motion of $\mathcal{L}$ is described by the scalar transport equation

$$\frac{\partial \mathcal{L}}{\partial t} + \boldsymbol{u} \cdot \nabla \mathcal{L} = 0, \tag{1}$$

where $\boldsymbol{u}$ is the velocity field. This equation is solved using a geometric semi-Lagrangian scheme, which is the same as the methodology used to transport the VoF representation of the phase interface [15, 16]. Using the same discretization for VoF and $\mathcal{L}$ transport ensures $\mathcal{L}$ moves with liquid parcels even near the phase interface where discontinuities in $\mathcal{L}$ exist.

The procedures described above provide $\mathcal{S}$ and $\mathcal{L}$ values in each computational cell. These identifiers are organized into lists by looping over the domain and storing only unique pairs of the $\mathcal{S}$ and $\mathcal{L}$, duplicate pairs are used to find the total volume of each of the structures. With these lists merges and splits are identified.

A merge occurs when a structure (single $\mathcal{S}$ value) has multiple $\mathcal{L}$ associated with it. The process of identifying merges is done by first sorting the lists by $\mathcal{S}$ and then searching the list for multiple $\mathcal{S}$, which by construction each have a different associated $\mathcal{L}$. When a merge case is identified the $\mathcal{L}$ of the new droplet is updated to the $\mathcal{L}$ associated with the largest structure that participated in the merge. Once all the merges have been identified for the current time step and the lists of IDs and volumes created, the lists are then exported to a graph database.

The identification of splits is similar to merges except the lists are sorted by $\mathcal{L}$ instead of $\mathcal{S}$. When multiple structures have the same $\mathcal{L}$ a split has occurred. The process of identifying splits is done by scanning through the sorted list of $\mathcal{L}$ looking for multiple structures with the same $\mathcal{L}$. Then the largest structure for this split case is identified. All other structures have split off this large structure and require new $\mathcal{L}$ values, which are assigned to the next largest available integer. When all of the splits for the current time step have been identified the lists are exported to a graph database. This process of identifying splits and merges is then repeated for the next time step.

Once all of the information for each time step has be sent to the graph database. The relationships between the droplets can be queried to show how the coherent liquid core breaks into many small droplets during atomization. The breakup dynamics include additional information including the structure shape, size, and flow field characteristics the structure existed in. The database creates a way to view the ancestry of the droplets and readily query the data to gather details on the atomization to test and improve lower order break up models.

Results and Discussion

Liquid Core Instability Extraction

The DMD analysis is performed on a three-dimensional simulation of an atomizing diesel type jet. The simulation is run with a mesh size of 64x64 in the spanwise directions and 256 in the axial direction. The interface is initialized as a half sphere. The inlet velocity field is defined using a stored velocity field from a previous simulation of a turbulent pipe flow. A Reynolds and Weber number of 25000 is used to have sufficient atomization. Note that with this high Reynolds and Weber number on the relatively coarse mesh, the results are not a DNS, but allow the DMD analysis to be tested. Data used in the DMD analysis is collected after the simulation reached a pseudo-steady state of atomization and the jet extended through the axial domain. This is acceptable because the goal was to capture the dynamics along the jets core and make a statement about how they affect the jet breakup. It can be seen in Fig. 2 that the jet starts to atomize within this domain and Fig. 3 shows $R(x, \theta)$ representation of the jet.

The domain captures both the jets core and the initial stages of atomization. The DMD analysis is performed on a slightly truncated version of Fig. 3. This is done to capture the jets core only more effectively, and more specifically to only capture the dynamics that are accurately represented by the DMD process. Since the DMD process uses a linear operator to approximate a non-linear system, once the system starts to become defined more dominantly by non-linear or random dynamics (the atomization towards the end of the jet) the DMD less accurately models the system. However, the DMD analysis is well suited to capture the initial growth of perturbations on the liquid core, which are likely linear.

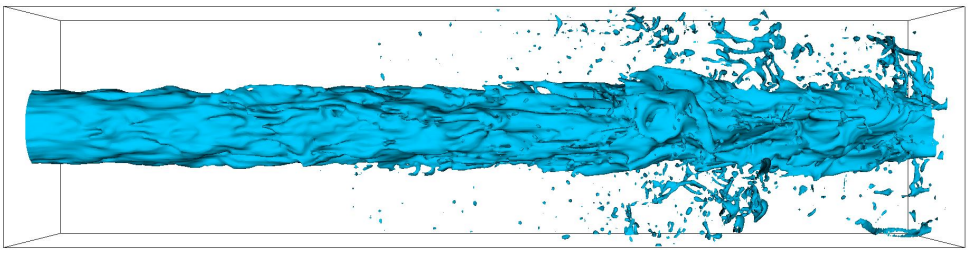

Figure 2. Rendering of gas-liquid interface of diesel type simulation at pseudo-steady state.

Performing the DMD on the truncated results lead to one primary dynamic mode that carries most of the energy of the system (shown in Fig. 4) and lots of smaller magnitude and repetitive dynamic modes. A few of those are shown in Figs. 5 and 6. The primary mode is nearly steady in time, while the lower magnitude modes are oscillatory. It might be intuitive to assume that the destabilization modes would be unstable, but since the the whole system is in a pseudo-steady state, the modes will not necessarily be blowing up. As a result, the oscillations in time dynamics decay slightly with time, while the magnitude of fluctuations grow over the axial domain. With the knowledge that jet breakup is occurring during the simulation, the non-steady time dynamics and growing amplitude towards the end of the axial domain indicate that the lower magnitude oscillatory modes, or some combinations of multiple repeating lower magnitude modes, are the driving force behind the jet breakup. When looking at jet breakup, the wavelength and frequency of the breakup modes are of interest. Breakup mode wavelength is indicative of separation droplet size while frequency is indicative of the rate at which the droplets are separated from the core. Comparing the 17th and 22nd modes, mode 17 has a larger wavelength and lower frequency than mode 22. This indicates that the 17th mode is contributing to larger droplets exiting the jets core than the 22nd mode, but mode 22 contributes to more frequent droplet separation.

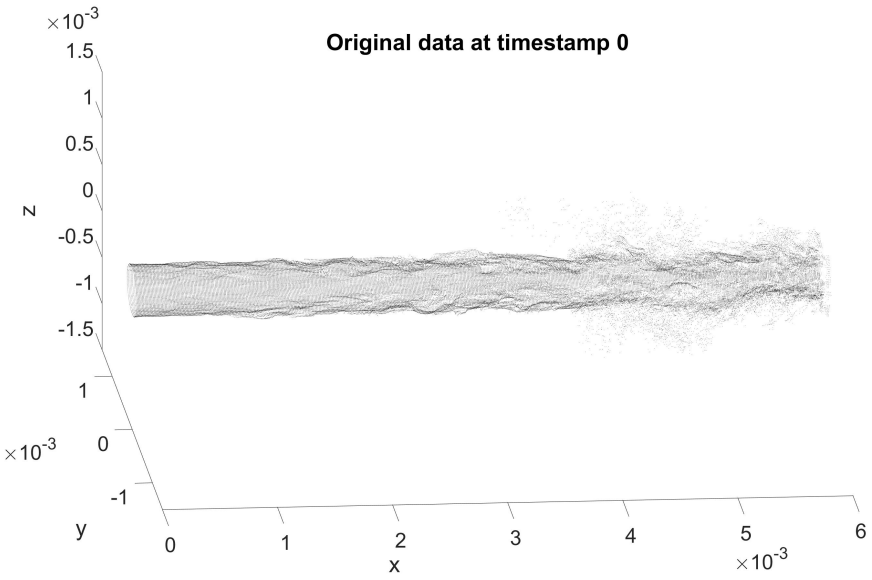

Figure 3. Representative $R(x, \theta)$ used in the DMD analysis.

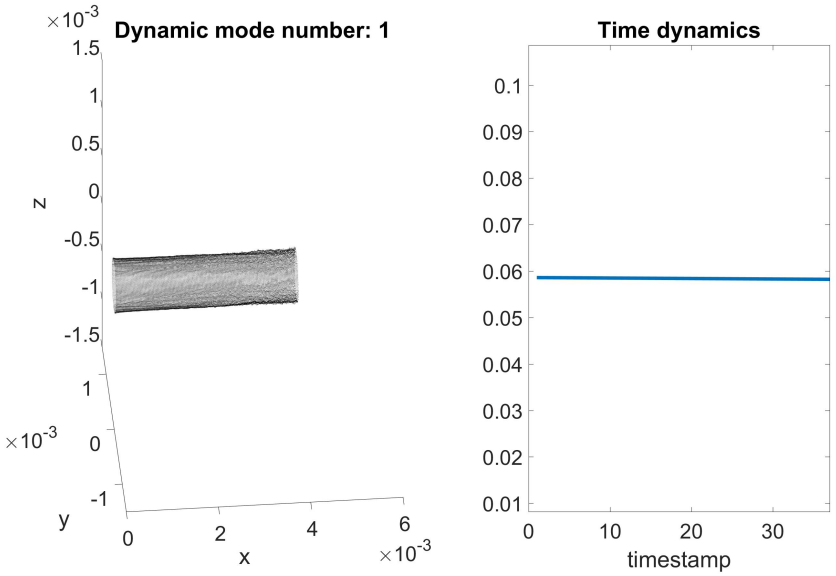

Figure 4. Primary dynamic mode and its time dynamics

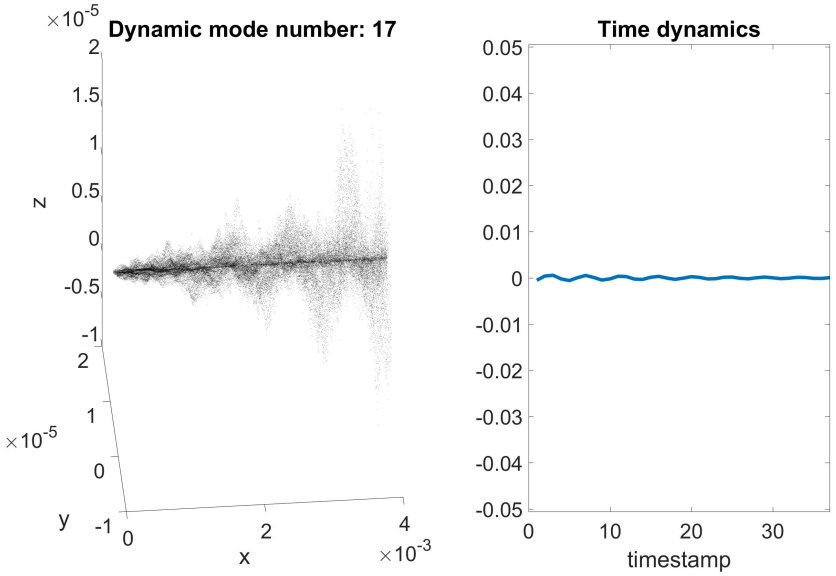

Figure 5. 17th dynamic mode and its time dynamics

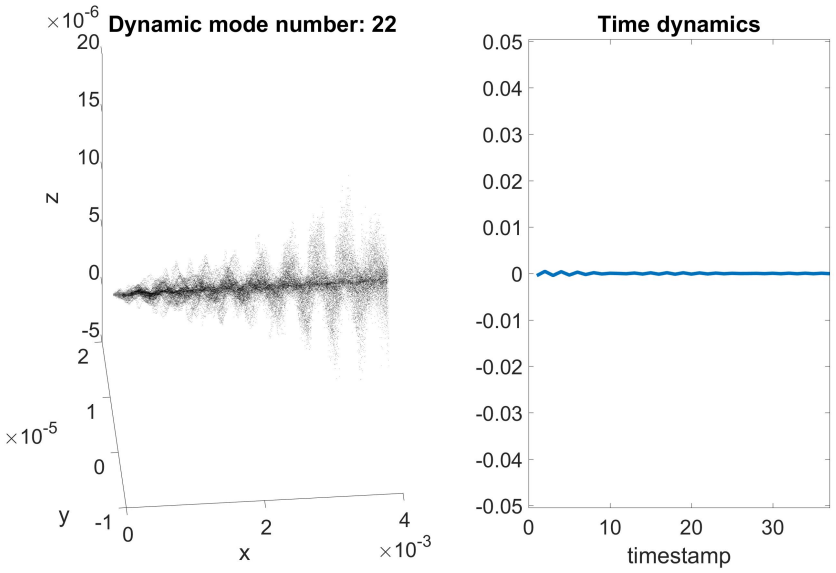

Figure 6. 22nd dynamic mode and its time dynamics

Droplet Ancestry Extraction Results

For the ease and efficiency of testing the proposed tool, a two-dimensional multiphase jet is created. The configuration is an air-blast atomizer consisting of slow moving liquid injected in a high velocity co-flow of air as shown in Fig. 7.. This case, while not physical, mimics realistic atomization and has breakup events that can be identified by the physics extraction tool.

To validate the droplet ancestry extraction tool several test cases were run with varying random fluctuations induced to the liquid core causing different break up patterns to form. Figure 8 shows a total of 9 different structures. The numbers located next to the droplets indicate the corresponding structure identification number $\mathcal{S}$. The output from the extraction tool indicates that there should be 9 total structures in the simulation at that given time which matches what is seen in the simulations output. Each structure also has its on $\mathcal{L}$ assigned to it by the droplets ancestry extraction tool that moves with the fluid structures between time steps. This can be seen in Fig. 9 with each individual structure has its own $\mathcal{L}$ indicated by the different colors of the structures. The corresponding

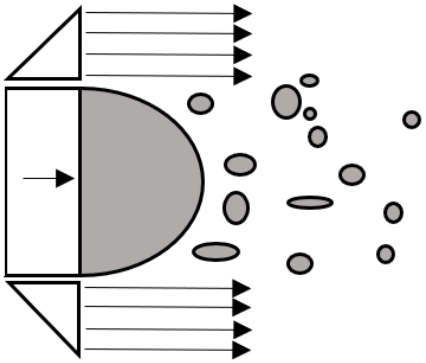

Figure 7. Illustration of the 2D multiphase jet configuration. Slow moving liquid is injected between high speed air that is injected above and below.

$\mathcal{L}$ value is written next to the structure. Currently, when a split or merge occurs within the simulation, the droplet dynamics ancestry tool correctly identifies the events and updates the $\mathcal{S}$ and $\mathcal{L}$ accordingly. All of this information is written to an output file that will be imported into a graph database along with statistics on the droplet shape, size, and a characterization of the local flow field the droplet existed in.

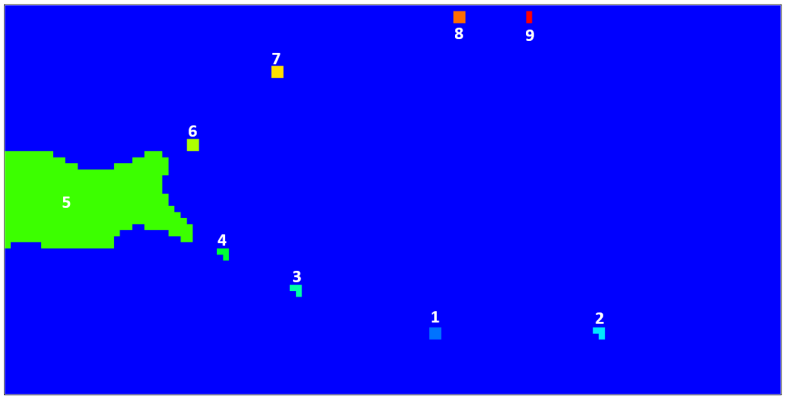

Figure 8. Structure identification number $\mathcal{S}$ at the given time step.

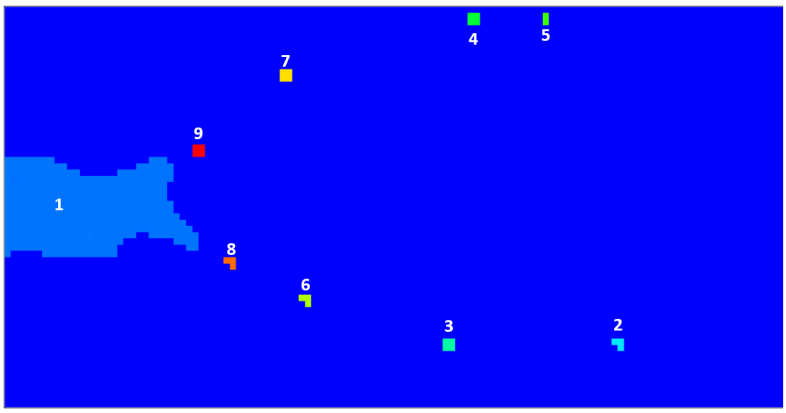

Figure 9. Liquid identification number $\mathcal{L}$ at the given time step.

Summary and Conclusions

In this paper, two physics extraction techniques are proposed to characterize the instability on an atomizing jet's liquid core and the ancestry of liquid structures produced in the spray. The tools reduce the complexity of results from DNS simulation providing information that could increase the accuracy of lower order models and gain a better understanding of what is happening within the atomization process. Future work is needed to completed the droplet ancestry extraction tool in regards to gathering more information for each droplet to gain a more in depth understanding of what is happening within the atomization process. The core instability tool has shown success in separating the non-time-varying dynamics of the system from the time-varying dynamics, indicating which set of dynamics are contributing to the jet breakup. The wavelengths and frequencies of the identified destabilization modes can be related to droplet size and exit frequency for implementation of a lower order sub-grid model. Further study on which of the identified jet breakup modes or possible combinations of them are most significant should be carried out. Also, more simulations using different parameters (Reynolds/Weber numbers) and types of jets should be explored with the intent of discovering a possible relationship between destabilization modes and certain parameters across jet types.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1511325 and Grant No. 1749779. Computational efforts were performed on the Hyalite High-Performance Computing System, operated and supported by University Information Technology Research Cyberinfrastructure at Montana State University.

References

[1] Yue Wang, Won Geun Lee, Rolf Reitz, and Ramachandra Diwakar. *SAE Technical Papers*, 2011.
[2] Christos Chryssakis and Dennis N. Assanis. *Atomization and Sprays*, 18(5):375–426, 2008.
[3] F.X. Tanner. *SAE Technical Papers*, Detroit, MI, United states, 1997.
[4] Valeri Kirsch, Manuel Armin Reddemann, Johannes Palmer, and Reinhold Kneer. *Atomization and Sprays*, 27(9):791–805, 2017.
[5] Marco Arienti and Marios C. Soteriou. *Physics of Fluids*, 21(11):112104, November 2009.
[6] Stanislav Kostka, Amy C. Lynch, Bethany C. Huelskamp, Barry V. Kiel, James R. Gord, and Sukesh Roy. *Combustion and Flame*, 159(9):2872–2882, September 2012.
[7] T. F. Fric and A. Roshko. *Journal of Fluid Mechanics*, 279(-1):1, November 1994.
[8] Shervin Bagheri, Philipp Schlatter, Peter J. Schmid, and Dan S. Henningson. *Journal of Fluid Mechanics*, 624:33, April 2009.
[9] P. J. Schmid, L. Li, M. P. Juniper, and O. Pust. *Theoretical and Computational Fluid Dynamics*, 25(1-4):249–259, June 2011.
[10] Peter J. Schmid. *Experiments in Fluids*, 50(4):1123–1130, April 2011.
[11] Peter J. Schmid, Daniele Violato, and Fulvio Scarano. *Experiments in Fluids*, 52(6):1567–1579, June 2012.
[12] Olivier Desjardins, Guillaume Blanquart, Guillaume Balarac, and Heinz Pitsch. *Journal of Computational Physics*, 227(15):7125–7159, July 2008.
[13] Haecheon Choi and Parviz Moin. *Journal of Computational Physics*, 113(1):1–4, July 1994.
[14] O. Desjardins, V. Moureau, and H. Pitsch. *Journal of Computational Physics*, 227(18):8395–8416, 2008.
[15] Mark Owkes and Olivier Desjardins. *Journal of Computational Physics*, 270(1):587–612, August 2014.
[16] Mark Owkes and Olivier Desjardins. *Journal of Computational Physics*, 332:21–46, March 2017.
[17] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. *Journal of Computational Physics*, 152(2):457–492, 1999.
[18] Mark Owkes, Eric Cauble, Jacob Senecal, and Aaron Currie. *Journal of Computational Physics*, 2017.
[19] J. Nathan Kutz, Steven L. Brunton, Dirk M. Luchtenburg, Clarence W. Rowley, and Jonathan H. Tu. *Journal of Computational Dynamics*, 1(2):391–421, December 2014.
[20] M. Herrmann. *Journal of Computational Physics*, 229(3):745–759, February 2010.