Parallel VOF Spray Droplet Identification in an Unstructured Grid

David A. Hebert[1], David P. Schmidt[1*], Darin A. Knaus[2],
Scott Phillips[2], Patrick J. Magari[2]

[1]Deptartment of Mechanical and Industrial Engineering, University of Massachusetts Amherst, Amherst, MA 01003 USA
[2]Creare Inc., Hanover, NH 03755 USA

Abstract

High-speed sprays operate on numerous length scales, providing a challenge to modelers who wish to directly simulate droplet formation. The smallest droplets set the lower bound on computational cell size while the size of the surrounding device determines the domain extent. For example, direct simulation of a spray in a jet engine requires cell sizes of a few microns and a domain size on the order of tens of centimeters to directly simulate all the interesting spray and combustion phenomena. It is neither efficient nor practical to attempt to calculate all length scales at once.

In contrast, Lagrangian particle tracking of spray droplets provides a very efficient and relatively low-cost method of modeling the spray. Droplet dynamics are adequately computed in the sparse spray, but near the intact liquid core the simulations can provide very inaccurate predictions. The ubiquitous neglect of volume occupied by the Lagrangian droplets in the gas phase equations is part of the problem. In the case of a spray in cross-flow, the gas tends to push through the Lagrangian spray instead of reacting to the liquid jet as a bluff body. The difficulties of modeling primary atomization compound the challenges.

One potential solution is to hybridize the two modeling approaches into a single composite spray modeling approach. The relatively large liquid jet and the primary atomization process are modeled with Volume of Fluids (VOF). The droplets are automatically identified as they are formed by a general recognition algorithm based on Connected Components Labeling (CCL). CCL is a well-known approach to image recognition and computer vision that is efficient and versatile. In the present work, CCL provides a bridge between VOF and Lagrangian particle tracking. The method is fully parallelized and can be used on any type of computational grid

* Corresponding author

Introduction

Sprays are inherently a multi-scale problem, with the smallest droplets on the order of microns and the spray typically evolving on the order of centimeters. Common spray modeling techniques include the volume of fluid (VOF) and discrete phase modeling (DPM) methods. Each of these strategies has particular strengths and weaknesses that must be taken into consideration with performing a spray simulation.

VOF utilizes a "color function" to signify the percent of a computational cell that is occupied by a particular fluid phase. This color function is implemented directly into the continuity equation, thus no *a priori* knowledge of the phase distribution or assumptions of phase behavior is required. However, in order to adequately resolve spray atomization, very high spatial resolution is required, resulting in large computational cost and limiting the model to a very small simulation domain.

In contrast, when using DPM it is assumed that the liquid phase does not occupy volume in a computational cell and reacts to forces induced from the main flow. While this may be a reasonable assumption downstream of the fuel injector, it certainly is not adequate near the injector where a significantly large portion of the computational cell (if not the entire cell) is liquid phase. Also, DPM requires an assumption about the initial spray droplet distribution for modeling primary atomization.

Given the way in which VOF and DPM are formulated, it is evident that the strength of VOF is in calculating details of the liquid core evolution while the strength of DPM is in the dispersed droplet evolution. In the modeling of a complete spray flow a "hybrid" method that utilizes both VOF and DPM would be advantageous (e.g., [1-2]). For example, one could use VOF in the vicinity of the fuel injector to directly model spray atomization and breakup. From the VOF portion, droplet statistics (e.g., mean diameter, mass distribution, etc.) could then be obtained and fed into a less computationally expensive DPM model. In this way both the small and large scale phenomena are computed without the need for arbitrary spray assumptions.

A hybrid VOF/DPM approach to spray modeling requires recognizing and quantifying the droplets and ligaments formed from the primary atomization process. This paper focuses exclusively on a versatile method to identify droplets formed in a VOF spray simulation. Droplet identification is performed accurately and efficiently using an algorithm based on connected components labeling (CCL) [3-4]. The methodology of CCL and algorithm parallelization are described in the following section, followed by simulation results for a spray in cross-flow.

Droplet Identification

Droplet identification employs a technique known as connected components labeling (CCL) [3,4] to efficiently identify adjacent pockets of liquid within the flow domain. A flowchart of the droplet identification algorithm is shown in Figure 1. Note the algorithm described here is for a serial (i.e., single processor) droplet identification application. An explanation of how the algorithm is made to run on parallel computers is described below. After checking if it is time to perform droplet identification, the first step in the identification algorithm is to initialize the entire domain

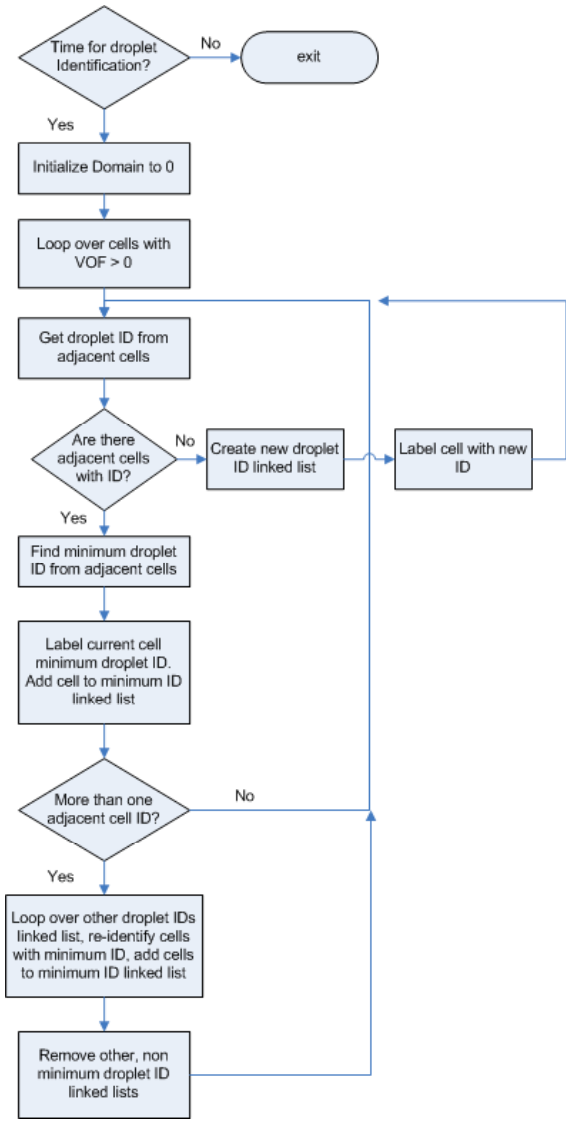


Figure 1: Schematic diagram of droplet ID algorithm

with zero droplet ID. A linked list of cells that contains liquid phase (i.e., cells with VOF > specified minimum value) is then created. The creation of a linked list of cells with VOF allows for efficient droplet identification since only cells of interest are used. Next, the routine loops over the linked list of VOF cells and obtains droplet ID's from all adjacent cells. This is illustrated in Figure 2 for both structured and unstructured two-dimensional grids for quadrahedral, triangular, and hybrid schemes. Three-dimensional cases follow directly, where the queried adjacent cells are all of those cells sharing faces with the cell in question. If none of the adjacent cells have been identified then the current cell is a new droplet; a new linked list for the new droplet is created and the cell is labeled with the new droplet ID. Otherwise, if any of the adjacent cells have a droplet ID then the current cell belongs to an existing droplet. The current cell is labeled with the minimum droplet ID of all the adjacent cells and then added to the corresponding droplet linked list. The routine then checks to see if there are multiple adjacent droplet ID's. If multiple adjacent droplet ID's exist then all adjacent droplets correspond to one droplet since they are connected via the current cell. These multiple droplets must be merged into one droplet. To merge the droplets the routine loops over the cells of droplets not labeled with the minimum ID, re-identifies each cell to the minimum droplet ID, and adds the cell to the minimum droplet ID linked list (Figure 3). The non-minimum ID droplet linked lists are then removed, resulting in a series of linked lists of cells corresponding to each droplet.

Algorithm Parallelization

Since simulations are becoming greater and greater in size (it is not uncommon for simulations to have on the order of 10 million cells), many spray simulations today are performed on large multiprocessor parallel computers. In fact, for such large simulations memory requirements demand that the simulation be run in parallel. In light of this, the droplet identification algorithm outlined above should also have the ability to be run in parallel, allowing it to be used in any size simulation. To distribute the problem among a number of processing elements (PE's) the computational domain is divided into a number of sub-domains, one for each PE. The droplet identification algorithm described above is then performed by each PE on its respective sub-domain. This will result in a list of droplet ID's ranging from 1 to N and local to each PE. At this point the droplet ID's are not unique since the ID's begin with the number 1 on each PE (Figure 4). For this application unique droplet ID's are required because the statistics of each droplet are of interest, and non-unique ID's will lead to confusion when statistics are computed for each droplet. In order to obtain unique droplet ID's the fact that each PE has a local ID ranging from 1 to N is utilized. A unique ID for each droplet can be readily obtained by simply adding the number of droplets (N) from each lower PE to the current PE droplet ID. This process is demonstrated in Figure 5 for a 4 PE simulation.

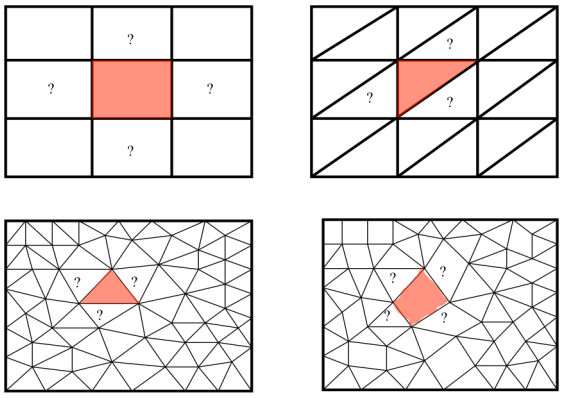


Figure 2: Example of adjacent cell query for VOF over several grid types

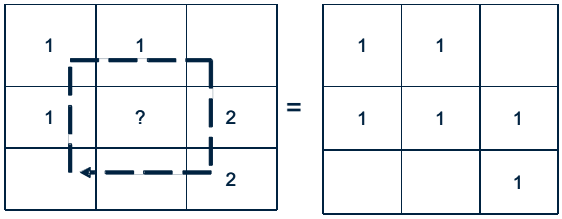
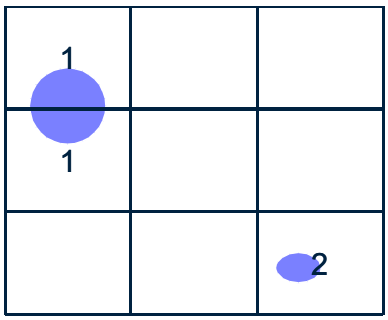
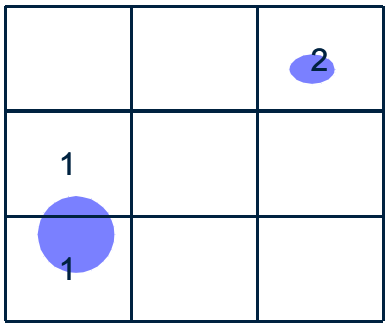


Figure 3: Merging of several previously identified droplets into one coherent droplet.



Sub-Domain 1



Sub-Domain 2

Figure 4: Illustration of local numbering scheme on each PE's sub-domain

PE 0 N=4 Local ID 1-4	PE 1 N=3 Local ID 1-3
PE 2 N=6 Local ID 1-6	PE 3 N=2 Local ID 1-2
PE 0 N=4 Unique ID 1-4	PE 1 N=3 Add 4 Unique ID 5-7
PE 2 N=6 Add 4+3=7 Unique ID 8-13	PE 3 N=2 Add 4+3+6=13 Unique ID 14-15

Figure 5: (Top) Each PE with a local droplet identification list. (Bottom) Each PE adds the number of droplets (N) from lower PE's, creating a unique ID across all PE's

Sub-domain Boundary Considerations

As the simulation progresses in time it is possible that a droplet will cross a PE boundary and be located on multiple PE's at one instant in time (Figure 6). Failing to account for liquid that spans multiple PE's would effectively split the droplet along the boundary between PE's. This will result in the creation of smaller droplets on each PE rather than one large droplet that spans PE's. It is necessary to merge the two droplets back into one large droplet to ensure that the resulting statistics will not be biased toward smaller sized droplets.

In order to merge droplets across PE's the concept of "halo" cells is used. In a parallel simulation, each PE is given a list of "interior" and "exterior" cells. Interior cells are those cells that are partitioned to that particular PE (Figure 7 solid cells). Exterior cells are cells that are interior to other PE's but are on the boundary with the current PE (Figure 7 dashed cells). In the droplet identification process, a cell that is exterior on one PE may be labeled differently than on the PE where the cell is interior (Figure 7, Step 1). The first step in connecting a droplet across PE's is to exchange the information on exterior cells (Figure 7, Step 2). That is, the droplet ID for an exterior cell is swapped with the ID from the PE where the cell is interior. The routine then loops through the linked list of cells for each droplet and checks to see if the new exterior cell ID is different than the interior ID. If the new exterior droplet ID is greater than the interior ID, the exterior ID is changed to the ID of the interior cells. If the new exterior ID is less than the interior ID, all the interior cells are re-identified with the exterior ID (Figure 7, Step 3).

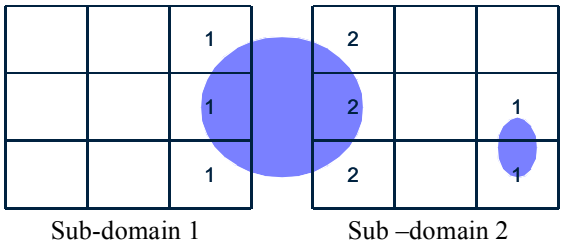


Figure 6: Droplet spanning multiple PE's

Results

A VOF simulation of a spray in cross-flow is used to demonstrate the droplet ID routine (Figure 8). The simulation domain is 10mm x 15mm x 6mm in the streamwise, height, and spanwise directions. The ambient air is initialized with a velocity of 75m/s and a temperature of 1000°C. Liquid phase jet-a fuel is injected through a 1mm diameter injector at 7.4m/s and 30°C. An unstructured grid is used with the longest edge of any computational grid cell 50μm, resulting in a domain with approximately 7.5 million cells. A sample is shown in the lower plot of Figure 8.

FLUENT® computational fluid dynamics (CFD) software is used for all numerical simulations in this study. The simulation is conducted using 16-32 PE's depending on license availability. The simulation is time-stepped for approximately 3.4ms, requiring approximately 15,000 CPU hours. The droplet identification algorithm is implemented via FLUENT® user defined function (UDF) and is hooked at the end of a time-step. Also, since Fluent uses "halo" cells in parallel applications, the droplet ID routine is easily implemented as described above.

In FLUENT® one has the ability to define multiple fluid "zones". Using this feature allows one to define a region in which the droplet identification is to be performed, reducing the number of grid elements on which the identification algorithm is performed. In this example, defining a separate fluid zone near the domain exit reduces the number of elements the ID algorithm is to be performed on from approximately 7.5 million to 2.9 million (Figure 8).

Droplet identification is performed in parallel using 16 PE's in the exit region of this domain. Figure 9 contains plots of the fuel spray and resulting droplet identification for the spray simulation at time 3.4ms. In the lower plot of Figure 9 it is seen that the droplet ID routine correctly identified all coherent droplets in the identification zone. Any difference in droplet size is between the figures is due to isosurface generation. The identification algorithm is very efficient, taking only 2.2 seconds to complete, compared with the approximately 7.3 minutes per time-step at 16 PE's. Hence, the droplet ID algorithm takes less than 0.5% of the overall simulation time.

Summary

In this paper an accurate and efficient method of identifying droplets in a VOF spray simulation is presented. The recognized droplets can then be tabulated for off-line analysis or used as the source of discrete Lagrangian droplets in a DPM simulation. The VOF droplet identification algorithm is based on the connected components method to efficiently identify

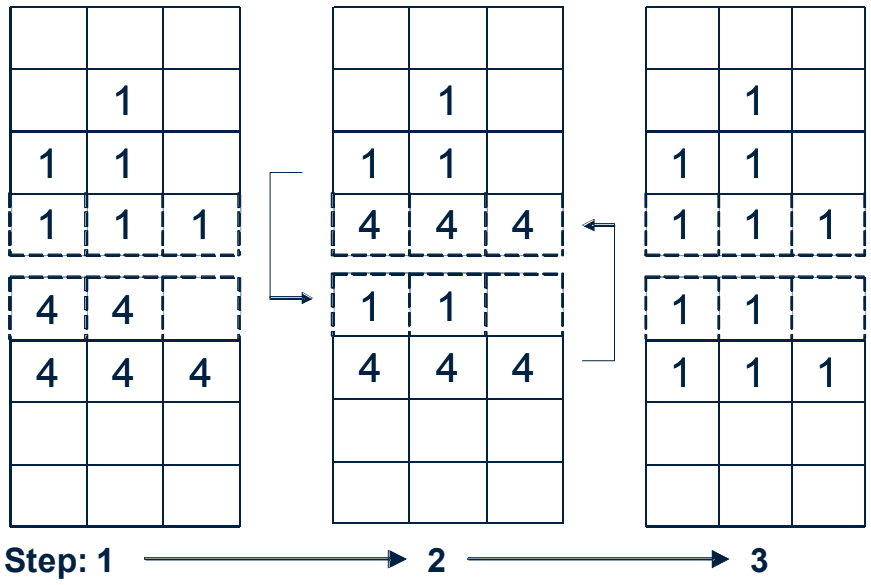
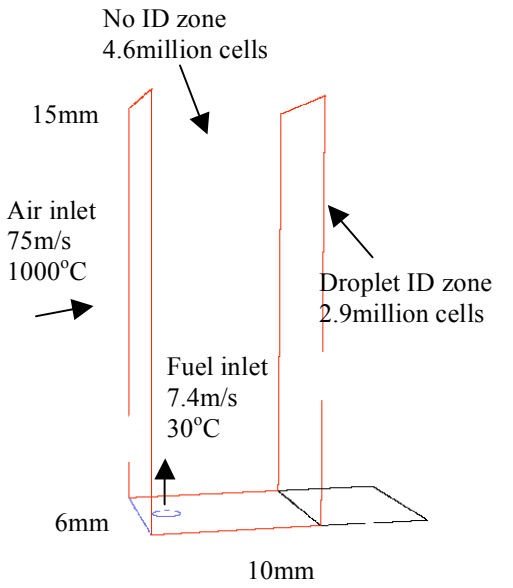


Figure 7: Identification and merging of a droplet spanning multiple subdomains



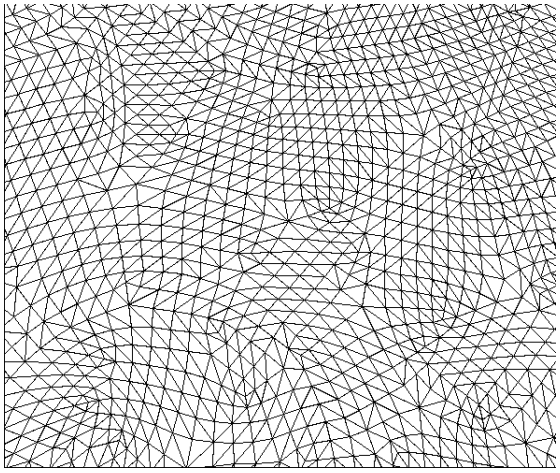


Figure 8: (Upper) Computational domain (Lower) sample of the computational grid.

neighboring cells that contain a specified amount of VOF. The algorithm can be used with any type of structured or unstructured grid.

The droplet ID algorithm can be executed in parallel and is not limited in the number of PE's that can be used for processing. The algorithm is parallelized through the use of halo cells that surround each PE's sub-domain allowing droplets that cross PE boundaries to be merged. Also, the algorithm can be performed only on a specified exit sub-domain, further increasing the efficiency in which the algorithm can be performed.

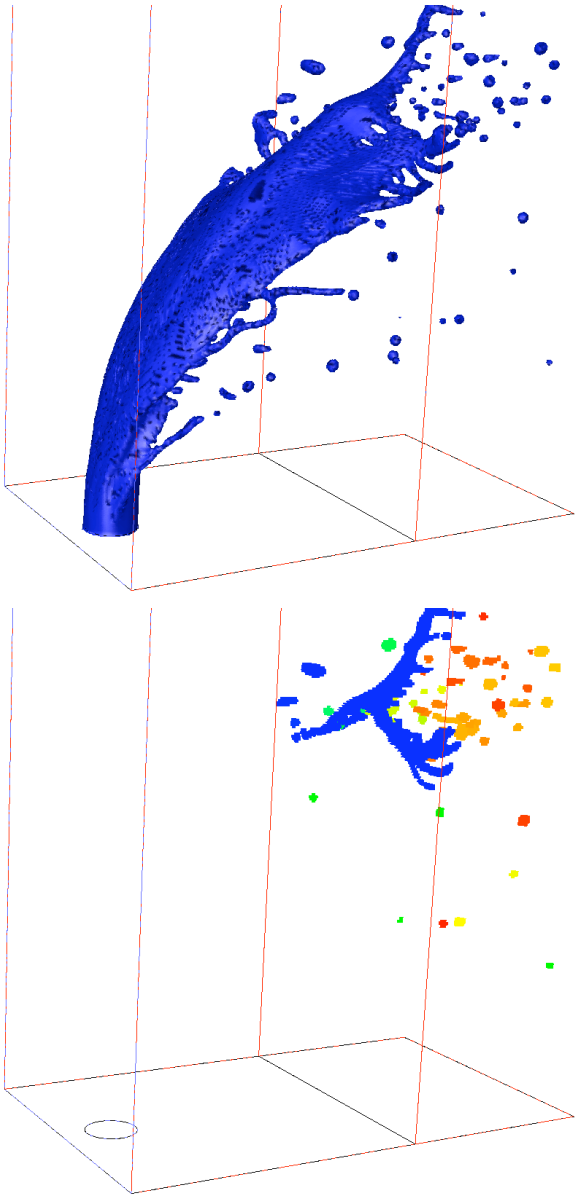


Figure 9: (Upper) Zoomed in liquid volume isosurface. (Lower) Droplet ID results using 16 PE's. The right sub-volume is specified for droplet identification.

It is shown that the algorithm can accurately identify droplets formed in a spray simulation. In a parallel simulation with approx. 7.5 million cells that takes over 7 minutes/time-step droplet identification can be performed in just over 2 seconds, which is nearly 0.5% of the runtime.

Acknowlegements

The authors wish to thank Dr. Fritz Bedford for help in adapting the droplet ID algorithm into a FLUENT® UDF and Dr. John Widmann for his advice

in running FLUENT® VOF simulations. This research was supported in part by the National Science Foundation through TeraGrid resources provided by the National Center for Supercomputing Applications (NCSA), University of Illinois Urbana-Champaign, grant number CTS060044T.

References

1. Bianchi, G. M., P. Pelloni, S. Toninel, D. Paganelli, and D. Suzzi. (2004). "A 2D Simulation Method For Computing Droplet Size Spectrum During The Atomization Of High-Speed Liquid Jets." Proc. ICEF04. Long Beach, California
2. Suzzi, D., E. V. Berg, J. V. Pastor, G. M. Bianchi, and R. Tatschl. (2004) "Simulation of Primary Break-up of Diesel Jets by a Hybrid Method Combining VOF-Calculations and the Classical DDM Rate Approach with a 3D CFD Code." ILASS-Europe. Nottingham
3. Han, Y., and Wagner, R.A., "An Efficient and Fast Parallel-Connected Component Algorithm," *Journal of the Association for Computing Machinery*. Vol. 37, No. 3. July 1990, pp.626–642.
4. Fisher, R., Perkins, S., Walker, A., and Wolfart, E., (2003), *<http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm>*