

multiUQ: An intrusive stochastic multiphase flow solver

B. Turnquist*, M. Owkes

Department of Mechanical and Industrial Engineering, Montana State University, U.S.A.

brian.turnquist@msu.montana.edu and mark.owkes@montana.edu

Abstract

Assessing the effects of input uncertainty on simulation results for multiphase flows will allow for more robust engineering designs and improved devices. For example, in atomizing jets, surface tension plays a critical role in determining when and how coherent liquid structures break up. Uncertainty in the surface tension coefficient can lead to uncertainty in spray angle, drop size, and velocity distribution. Uncertainty quantification (UQ) determines how input uncertainties affect outputs, and the approach taken can be classified as non-intrusive or intrusive. A classical, non-intrusive approach is the Monte-Carlo scheme, which requires multiple simulation runs using samples from a distribution of inputs. Statistics on output variability are computed from the many simulation outputs. While non-intrusive schemes are straightforward to implement, they can quickly become cost prohibitive, suffer from convergence issues, and have problems with confounding factors, making it difficult to look at uncertainty in multiple variables at once. Alternatively, an intrusive scheme inserts stochastic (uncertain) variables into the governing equations, modifying the mathematics and numerical methods used, but possibly reducing computational cost. In this work, we extend UQ methods developed for single-phase flows to handle gas-liquid multiphase dynamics by developing a stochastic conservative level set approach and a stochastic continuous surface tension method. An oscillating droplet and a 2-D atomizing jet are used to test the method. In these test cases, uncertainty about the surface tension coefficient and initial starting position will be explored, including the impact on breaking/merging interfaces.

Introduction

multiUQ is a novel stochastic multiphase flow solver which utilizes an intrusive UQ scheme to solve the stochastic incompressible Navier-Stokes equations. The objective of this project is to improve upon non-intrusive methods, such as the well known Monte-Carlo [1], which are simple to deploy upon a pre-built deterministic solver, but suffers from convergence issues and computational cost [2]. Where non-intrusive methods can be quickly deployed on an existing solver, an intrusive scheme requires changing the fundamental governing equations by substitution of stochastic (random) variables, which vary in uncertainty dimensions.

While there are several types of spectral expansions that allow a variable to vary in uncertainty space, this project utilizes a polynomial chaos method, as first described by Wiener [3], and implemented in a single phase, incompressible flow solver by Le Maître et al. [4], among others ([5], [6], [7], [8]). For this approach, any stochastic random variable y is taken to be a function of any number of P orthogonal polynomial basis functions, such that

$$y(\mathbf{x}, t, \boldsymbol{\zeta}) = \sum_{k=0}^P y_k(\mathbf{x}, t) \phi_k(\boldsymbol{\zeta}), \quad (1)$$

where $\mathbf{x}$ is the spatial directions, t is time, and $\boldsymbol{\zeta}$ is the stochastic dimensions. $\phi(\boldsymbol{\zeta})$ are orthogonal polynomials defined in uncertainty dimensions $\boldsymbol{\zeta}$. The result is that any variable then becomes a function of space, time, and uncertainty, among any other dependencies. In this work, the Legendre polynomials were utilized for simplicity, as well as their suitability in representing a uniform distribution. Some research has noted that the orthogonal basis functions used may be important for describing the distributions present [9]. Legendre polynomials are orthogonal, i.e.,

$$\int_{-1}^1 \phi_k \phi_b \delta \zeta = \begin{cases} \langle \phi_k \phi_b \rangle & k = b \\ 0 & k \neq b \end{cases} \quad (2)$$

where multiplication by a test function ϕ_b and integration results in zero with any basis function other than the special case where $k = b$. The orthogonality property simplifies the stochastic governing equations and reduces

*Corresponding author: brian.turnquist@msu.montana.edu

computational cost. Other orthogonal polynomial basis functions, i.e., Hermite polynomials, could be readily used in this framework.

Having discussed the method of spectral expansion used to represent stochastic variables, we now consider how random variables are incorporated into the governing equations. The deterministic Navier-Stokes equations for incompressible flow (used for multiphase) can be written as

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{\nabla P}{\rho} + \nu \nabla^2 \mathbf{u} + \frac{\mathbf{f}_\sigma \delta_s}{\rho}, \quad (3)$$

where the velocity at any given point in time and space is a function of viscosity ν , pressure gradient ∇P , density ρ , and surface tension force $\mathbf{f}_\sigma \delta_s$.

Claiming uncertainty exists about one variable requires uncertainty to exist about at least one other. For a gas-liquid multiphase flow, uncertainty about velocity effects the location of the phase interface leading to uncertainty about density, viscosity, surface tension and pressure.

The stochastic Navier-Stokes equations result from substituting random variables $\mathbf{u}(\zeta)$, $\rho(\zeta)$, $P(\zeta)$, and $\nu(\zeta)$ into the deterministic equation shown above. Multiplying by a test function, integrating over ζ , and solving for the basis weight of velocity $\mathbf{u}_b$ leading to

$$\frac{\partial \mathbf{u}_b}{\partial t} + \mathbf{u}_k \cdot \nabla \mathbf{u}_l C_{klb} = -\rho_k^I \nabla P_l C_{klb} + \nu_k \nabla^2 \mathbf{u}_l C_{klb} + \rho_k^I \mathbf{f}_{\sigma l} \delta_s C_{klb}, \quad (4)$$

where the inverse of density $1/\rho = \rho^I$ and C_{klb} is the multiplication tensor defined with

$$C_{klb} = \frac{\langle \phi_k \phi_l \phi_b \rangle}{\langle \phi_b \phi_b \rangle} = \frac{\int_{-1}^1 \phi_k \phi_l \phi_b d\zeta}{\int_{-1}^1 \phi_b \phi_b d\zeta}. \quad (5)$$

This tensor needs to be calculated only once at the beginning of each simulation, and acts as a scalar value which weights the interaction of the basis functions.

A stochastic continuity equation is obtained by substitution of $\mathbf{u}(\zeta)$ into the standard deterministic continuity equation ($\nabla \cdot \mathbf{u} = 0$), multiplication by a test function and integration over ζ . Due to the orthogonality of the basis functions, this reduces to the divergence of each velocity basis weight $\mathbf{u}_b$ must be divergence free, i.e.,

$$\nabla \cdot \mathbf{u}_b = 0. \quad (6)$$

Finally, a surface tension force must be calculated for a gas-liquid flow. A deterministic surface tension can be written as

$$\mathbf{f}_\sigma \delta_s = \sigma \kappa \mathbf{n} \delta_s, \quad (7)$$

where σ is the surface tension coefficient, κ is the curvature of the interface, and $\mathbf{n}$ represents the outward normal vectors at the interface boundary. The surface tension force exists only at the interface, as denoted by the addition of the Dirac delta δ_s . In a stochastic formulation, this interface location is allowed to vary, which requires a distribution of interfaces smoothed across space. The curvature, which is used to calculate the magnitude of the surface tension force is also stochastic and can be written as $\kappa_b = -\nabla \cdot \mathbf{n}_b$. Assuming there can be uncertainty about the surface tension coefficient σ , as well as the outward normal vector and curvature, the calculation of basis weight b of the surface tension force is found with

$$\mathbf{f}_{\sigma, b} \delta_s = \sigma_k \kappa_l \mathbf{n}_m \delta_s C_{klmb}, \quad (8)$$

where the multiplication tensor

$$C_{klmb} = \frac{\langle \phi_k \phi_l \phi_m \phi_b \rangle}{\langle \phi_b \phi_b \rangle}. \quad (9)$$

Numerical Methods

Implementation of the intrusive framework discussed required modifications of several published numerical methods, particularly with respect to interface transport and surface tension calculation. A modified conservative level set scheme is proposed, modeled after the work published by Desjardins et al. [10], which expanded upon

work by Olsson & Kreiss [11], among others ([12], [13], [14]). In this scenario, a signed distance function $g(\mathbf{x}, t)$ is transformed by way of a sigmoid to a smoothed Heaviside about the interface using

$$\Psi(\mathbf{x}, t) = \frac{1}{1 + \exp(-2g(\mathbf{x}, t)/\varepsilon)}, \quad (10)$$

where $\varepsilon = \max(2\Delta x, 2\Delta y)$. The level set transport for a stochastic level set is written as

$$\frac{\partial \Psi_b}{\partial t} + \nabla \cdot (\mathbf{u}_k \Psi_l) C_{klb} = 0. \quad (11)$$

Time marching is accomplished using an iterative semi-implicit Crank Nicolson method for stability, while spatial derivatives are calculated using a flux method from one side of the cell to the other to maximize mass conservation. Level set values Ψ_k are interpolated to the left and bottom cell walls utilizing a HOU-5 scheme such as that presented in [15], which has been shown to work effectively and efficiently for scalar transport.

Nothing ensures the shape of the conservative level set will be preserved throughout the transport process, so an additional reinitialization step is introduced. This is where the bulk of the modifications have been made with respect to the methodology published by Desjardins et al. [10]. In the original reinitialization equation, the level set is updated with

$$\frac{\partial \Psi}{\partial \tau} + \nabla \cdot (\Psi(1 - \Psi)\mathbf{n}) = \nabla \cdot (\varepsilon(\nabla \mathbf{n})\mathbf{n}), \quad (12)$$

where a unit normal vector $\mathbf{n}$, was used. In the present framework, the condition of unit magnitude was relaxed and a continuous normal vector $\mathbf{r}$ was introduced which can be easily incorporated into the stochastic equations. This vector is calculated using

$$\mathbf{r} = \frac{\nabla \Psi}{|\nabla \Psi|_{\max}}, \quad (13)$$

where the denominator scales the vectors based on the maximum gradient of the conservative level set profile. This gradient can be found analytically to be $|\nabla \Psi|_{\max} = 1/2\varepsilon$. Using a continuous normal vector on the reinitialization equation required the addition of a general diffusion term to smooth oscillations in the level set function away from the interface.

The stochastic level set reinitialization equation is derived by substitution of stochastic random variables $\Psi(\zeta)$ and $\mathbf{r}(\zeta)$, multiplication by a test function, and integration over ζ leading to the equation

$$\frac{\partial \Psi_b}{\partial \tau} = -\nabla \cdot (\Psi_k \mathbf{r}_l C_{klb} - \Psi_k \Psi_m \mathbf{r}_l C_{klmb}) + \nabla \cdot (\varepsilon_1 (\nabla \Psi_k \cdot \mathbf{r}_l) \mathbf{r}_m) C_{klmb} + \nabla \cdot (\varepsilon_2 \nabla \Psi_b), \quad (14)$$

which is advanced over pseudo-time using the same semi-implicit Crank Nicolson as the scalar transport scheme for stability. Additionally, gradients are calculated across the cell using a finite volume method to improve mass conservation, while the continuous normal vectors are found using a planar least squares approach.

Calculation of a time step for pseudo-time marching is dependent on the maximum analytic gradient of the level set scalar ($|\nabla \Psi|_{\max}$), the minimum cell width ($h_{\min} = \min(\Delta x, \Delta y)$), and the maximum current gradient of the level set for any basis function. For a CFL number of less than 1

$$\Delta \tau = \frac{\text{CFL}_{\text{reinit}} h_{\min} |\nabla \Psi|_{\max}}{3 \max(|\frac{\partial \Psi_b}{\partial x}|, |\frac{\partial \Psi_b}{\partial y}|)}. \quad (15)$$

The surface tension force is implemented based on the approach presented by Tryggvason et al. [16]. In this approach the interface normal vector is computed using

$$\mathbf{n} = \frac{\nabla \Psi}{|\nabla \Psi|} \quad (16)$$

and curvature the curvature is computed using

$$\kappa = -\nabla \cdot \mathbf{n}. \quad (17)$$

Extending these equations for a stochastic level set requires computing stochastic normal vector $\mathbf{n} = \mathbf{n}_k \phi_k$, where the basis weights are computed with

$$\mathbf{n}_b = \frac{1}{\langle \phi_b \phi_b \rangle} \int_{-1}^1 \frac{\nabla \Psi_k \phi_k}{\sqrt{(\partial \Psi_l \phi_l / \partial x)^2 + (\partial \Psi_m \phi_m / \partial y)^2}} \phi_b d\zeta. \quad (18)$$

This integral does not have an analytic solution and cannot be pre-computed due to the dependence on the level set weights. To evaluate the integral a Gaussian quadrature scheme is used. The other integrals, manifest in the C_{klb} and C_{klmb} tensors, can be pre-computed using analytic integrals of the basis functions. The lack of an analytic solution to Eq. 18 is a strong motivation for further research into the calculation of a stochastic curvature and surface tension.

Using the calculated unit normal vector, a stochastic curvature can then be calculated with

$$\kappa_b = -\nabla \cdot \mathbf{n}_b. \quad (19)$$

On a staggered grid, with velocity vectors held at cell faces, the divergence and curvature are located at the cell center using a simple difference scheme over the cell.

With a simple interpolation to the cell wall for the curvature, and utilizing the level set scalar Ψ in a modification to the method presented in [16], the density is calculated such that, for internal fluid density ρ_i and external fluid density ρ_e ,

$$\rho_b = \rho_{i,b} + \rho_{e,b} - \Psi_k \rho_{e,l} C_{klb}, \quad (20)$$

and the final smoothed calculation of surface tension is made using

$$\mathbf{f}_{\sigma,b} = \frac{\sigma_k \kappa_l \nabla \rho_m}{[\rho]_d} C_{klmb}, \quad (21)$$

where the gradient of density $\nabla \rho$ is found with a 4th order central difference scheme, and $[p]_d = \rho_i - \rho_e$.

Additionally, time marching for the velocity updates to the Navier-Stokes equations was done using a pressure correction method, where a semi-implicit Crank-Nicolson update advanced the predictor step and an elliptic solver was used to evaluate the pressure field and then correct. Determination of the time step for this case is, for a convective CFL less than 1,

$$\Delta t = \frac{\text{CFL} h_{\min}}{\nu \mathbf{u}_{\max}}. \quad (22)$$

Results and Discussion

2-D Oscillating Droplet

The oscillating droplet is used to assess the interactions of the stochastic surface tension force with the dynamics of the droplet. Here, an elliptic droplet, is initialized in a large undisturbed pool. Surface tension forces drive fluid flow causing the droplet oscillate. Analytic results were first suggested by Lord Rayleigh [17], where the period of oscillation,

$$\omega = 2\pi \sqrt{\frac{(\rho_i + \rho_e) R^3}{6\sigma}}, \quad (23)$$

is found to be a function of surface tension (σ), density (ρ_i and ρ_e), and unperturbed radius (R). Tracking the oscillations was done using a calculation of the global kinetic energy,

$$\text{KE}_g = \frac{1}{2} \int \rho(\mathbf{u} \cdot \mathbf{u}) \, dx \, dy, \quad (24)$$

where the kinetic energy peaks and ebbs as the ellipse oscillates in alternating directions.

The deterministic oscillating droplet is frequently used, such as by Garrick et al. [18], to test the abilities of a flow solver since the results can be compared to the analytic solution for the period of oscillation. Figure 1 depicts two time steps for a deterministic case, showing the velocity field and phase interface.

All fluid properties, initial and boundary conditions are assumed known. The period of oscillation for the case presented is shown in Fig. 2. The surface tension coefficient is $\sigma = 30 \text{ dyn/cm}$, and dimensions of the semimajor and

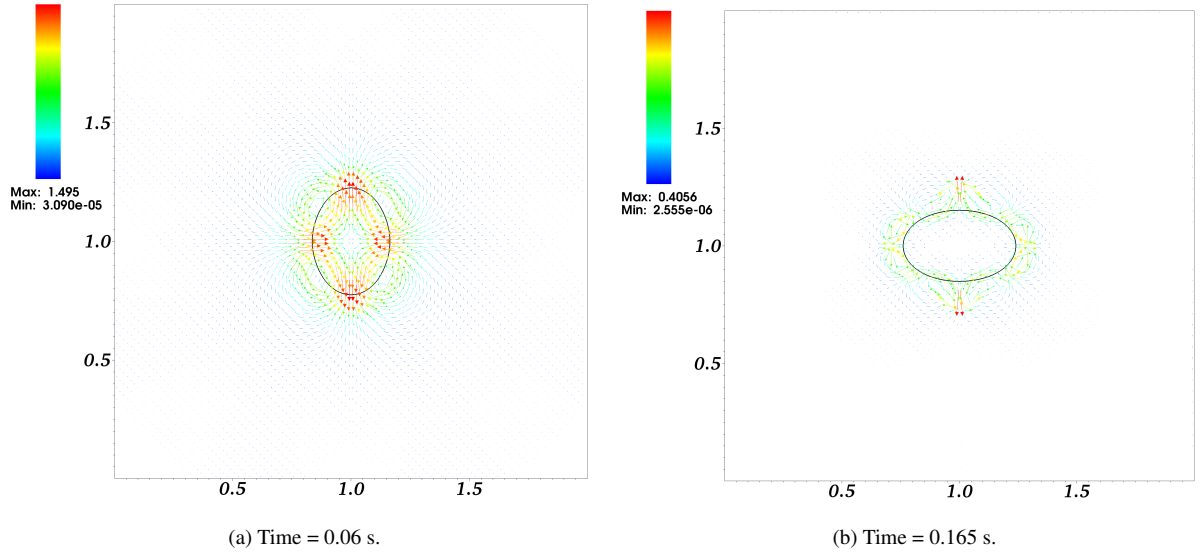


Figure 1: Time slices of a deterministic oscillating droplet on a 2×2 cm grid with a mesh size of 128×128 . Black lines denote the interface boundary and the color map is a function of velocity magnitude (cm/s).

semiminor axis of the ellipse are 0.25 and 0.15 cm, respectively. Internal and external fluid density are $\rho_i = 1 \text{ g/cm}^3$ and $\rho_e = 0.2$, respectively. Internal and external kinematic viscosity are $\nu_i = 0.01 \text{ cm}^2/\text{s}$ and $\nu_e = 0.00018$, respectively. With these inputs, the period of oscillation is $\omega = 0.047$. The axis of Fig. 2 is scaled by ω and there is good agreement on the period of oscillation, with some deviation. Because the droplet is oscillating faster than expected, it is suspected that the smoothing of the surface tension, which should exist only at the interface, has added extra energy to the system.

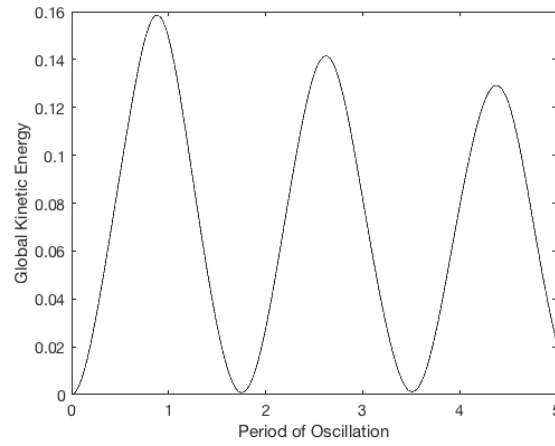


Figure 2: Global kinetic energy (1×10^{-7} Joules) for a deterministic oscillating droplet at 128×128 mesh. Time has been non-dimensionalized with t/ω .

Expanding upon the idea of the oscillating droplet, consider the possibility that the surface tension coefficient contains uncertainty. The uncertainty is represented by a uniform distribution such that the standard error about the mean is $\approx 33.3\%$ leading to $\sigma(\zeta) = 30 + 10\zeta$.

Fig. 3 displays two equally possible solutions. Notice that at the time slices shown, these droplets are offset, and the variance, illustrated with the color map, peaks where the droplets do not overlap. These areas of high variance indicate points where the uncertainty of being within the droplet are greatest.

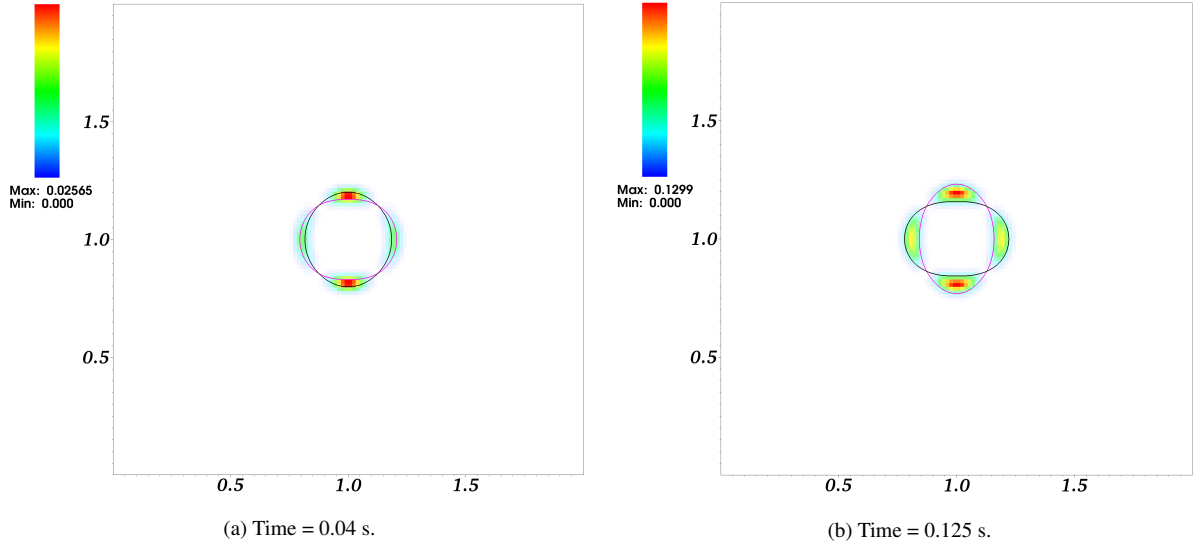


Figure 3: Time slices of a stochastic oscillating droplet with a mesh size of 128×128 . Uncertainty exists only about the surface tension coefficient. The color map illustrates the variance of Ψ (unitless). Lines depict interface for solution bounds, where magenta corresponds to $\sigma(\zeta = -1)$, and black corresponds to $\sigma(\zeta = +1)$.

2-D Atomizing Jet

For the case of an atomizing jet, it is again reasonable to begin by displaying a deterministic case, where all fluid parameters and initial and boundary conditions are known (equivalent to the droplet case). Figure 4 displays this case at 2 different time steps. In this case, the fluid is injected at a high velocity (100 cm/s), and mushrooming begins rather quickly. Eventually, the column begins to break up, releasing two droplets.

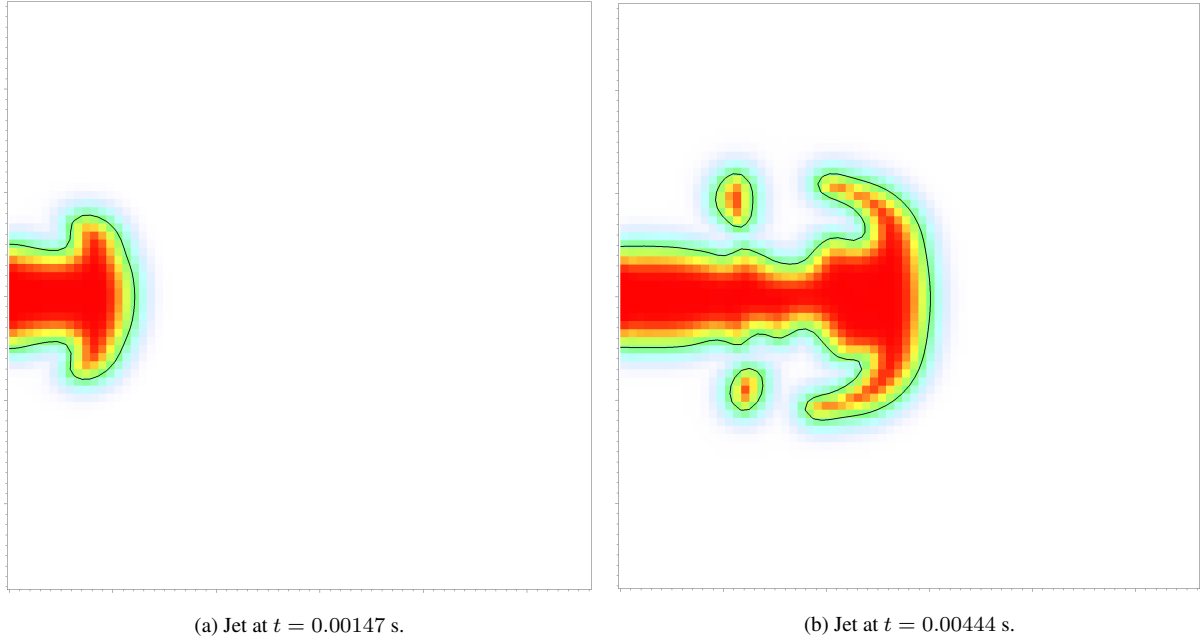


Figure 4: Zoomed image of a deterministic atomizing jet on a 128×128 mesh ($1 \times 1 \text{ cm}$ spatial grid). Black line denotes the interface boundary while the color map depicts regions of higher density ($\rho_i = \rho_{\max} = 1 \text{ g/cm}^3$, $\rho_e = 0.2$). Incoming jet velocity $u_{\text{in}} = 100 \text{ cm/s}$.

Consider again the possibility of uncertainty about the surface tension coefficient, where $\sigma(\zeta) = 30 + 10\zeta$. At one bounding solution, where $\sigma(\zeta = +1) = 40$, the surface tension is twice that of the other bounding solution,

where $\sigma(\zeta = -1) = 20$. Figure 5 presents the jet column at a given time for both of these equally possible solutions. The expectation here would be that a fluid that is injected with a much lower surface tension coefficient would begin to breakup much quicker than that of the fluid with the larger surface tension coefficient.

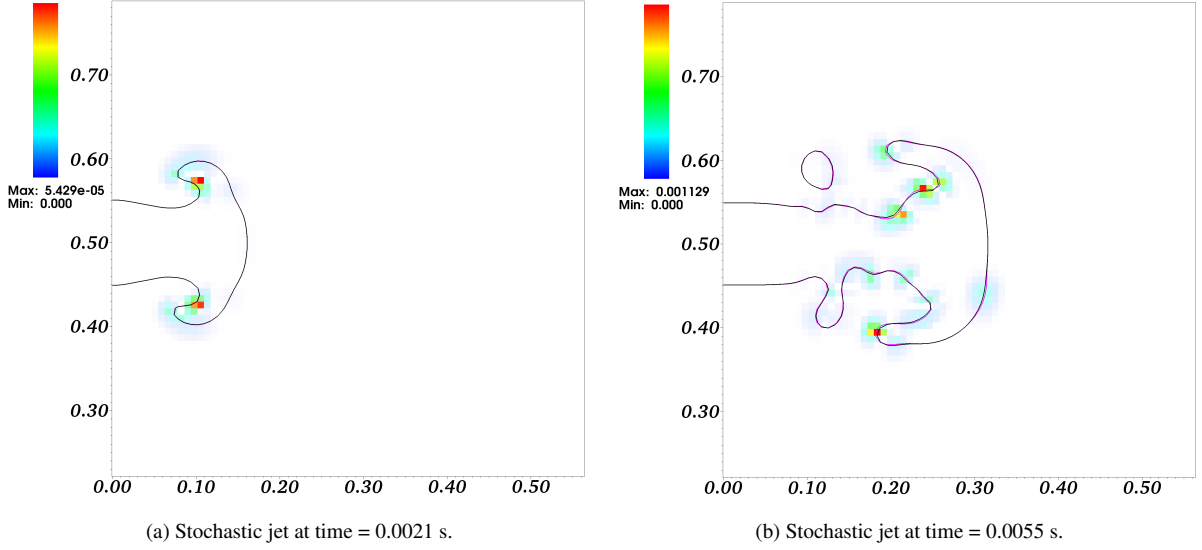


Figure 5: Zoomed image of a stochastic atomizing jet on a 128×128 mesh (1×1 cm grid), where uncertainty is assumed about the surface tension coefficient, σ . Magenta lines indicate the solution of $\sigma(\zeta = -1) = 20$ dyn/cm while black lines correspond to $\sigma(\zeta = +1) = 40$ dyn/cm. Color bars represent the variance about Ψ (unitless). Incoming jet velocity $u_{in} = 100$ cm/s.

Again looking at Fig. 5, the color map highlights regions of variance about the level set scalar Ψ . Areas of high variance indicate regions where the possibility of being within the injected fluid are more uncertain. It is easy to see that the difference between the two bounding solutions is slight, and likely is due to the velocity at which fluid is injected into the system. This is a surprising result, especially when compared to the results from the oscillating droplet. However, given the surface tension acts as a source term which is added to the velocity, as injected velocities are increased, the addition of the source term becomes proportionally smaller.

While only considering the effects of uncertainty about a single variable, this simulation provides some valuable insights. First, given the ability to solve for a range of solutions simultaneously, a better average jet breakup can be calculated. Second, rates of breakup will effect the droplet size and distribution and impact, e.g., combustion efficiency for engine applications, ultimately leading to more efficient combustion.

Summary and Conclusions

A mathematical and numerical framework for building an intrusive UQ gas-liquid multiphase direct numerical flow solver is presented in brief. With respect to the mathematical and numerical schemes outlined, some adjustments were made to previously published methods to incorporate the intrusive UQ method. A stochastic conservative level set transport is discussed, as well as a convergent reinitialization equation for maintaining a stochastic hyperbolic profile and associated mass conservation properties. Additionally, a method for calculation of a stochastic curvature and surface tension force are also presented.

Two multiphase test cases are discussed. In the first test case, that of the oscillating droplet, an analytic solution to the frequency of oscillation is reasonably well matched with that of that of the deterministic simulations as well as the simulations which assumed there to be uncertainty about the surface tension coefficient. The second case of an atomizing jet showed the proposed framework is able to predict uncertainty for more relevant application. Including a stochastic surface tension coefficient offered insights into the rate of column growth and the way in which breakup begins to occur.

A number of questions remain to be addressed to expand the methods presented here. Assessment of the number of basis function needed to adequately converge all variables present in the model is required. With this convergence information, the next step would be to compare the computational cost to a non-intrusive scheme like Monte-Carlo. Finally, the current curvature calculations require a Gaussian quadrature method, adding unnecessary error and expense to the method. Finding a quadrature-free curvature method suitable for a stochastic framework

will reduce numerical errors and decrease computational time.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1511325. 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] Nicholas Metropolis and Stanislaw Ulam. *Journal of the American Statistical Association*, 44(247):335–341, 1949.
- [2] Mary Kathryn Cowles and Bradley P. Carlin. *Journal of the American Statistical Association*, 91(434):883–904, 1996.
- [3] Norbert Wiener. *American Journal of Mathematics*, 60(4):897–936, 1938.
- [4] O.P. Le Maître, Matthew T. Reagan, Habib N. Najm, Roger G. Ghanem, and Omar M. Knio. *Journal of Computational Physics*, 181(1):9, September 2002.
- [5] O. P. Le Maître and Omar M. Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*. With Applications to Computational Fluid Dynamics. Springer Netherlands, Dordrecht, Dordrecht, 2010.
- [6] O.M. Knio and O.P. Le Maître. *Recent Topics in Computational Fluid Dynamics*, 38(9):616–640, September 2006.
- [7] O M Knio and O P Le Maître. *Fluid Dynamics Research*, 38(9):616, 2006.
- [8] O.P. Le Maître, Omar M Knio, Habib N Najm, and Roger G Ghanem. *Journal of Computational Physics*, 173(2):481–511, November 2001.
- [9] Dongbin Xiu and George Em Karniadakis. *Journal of Computational Physics*, 187(1):137–167, May 2003.
- [10] Olivier Desjardins, Vincent Moureau, and Heinz Pitsch. *Journal of Computational Physics*, 227(18):8395–8416, September 2008.
- [11] Elin Olsson and Gunilla Kreiss. *Journal of Computational Physics*, 210(1):225–246, November 2005.
- [12] Elin Olsson, Gunilla Kreiss, and Sara Zahedi. *Journal of Computational Physics*, 225(1):785–807, July 2007.
- [13] Mark Sussman, Peter Smereka, and Stanley Osher. *Journal of Computational Physics*, 114(1):146–159, September 1994.
- [14] Stanley Osher and Ronald P Fedkiw. *Journal of Computational physics*, 169(2):463–502, 2001.
- [15] R.R. Nourgaliev and T.G. Theofanous. *Journal of Computational Physics*, 224(2):836–866, June 2007.
- [16] G. Tryggvason, R. Scardovelli, and S. Zaleski. *Direct Numerical Simulations of Gas-Liquid Multiphase Flows*. Cambridge University Press, 2011.
- [17] F. R. S. Lord Rayleigh. *Proceedings of the Royal Society of London*, 29(196-199):71–97, January 1879.
- [18] Daniel P. Garrick, Mark Owkes, and Jonathan D. Regele. *Journal of Computational Physics*, 339:46–67, June 2017.