

Data-driven modeling for drop size distributions

Original

Data-driven modeling for drop size distributions / Traverso, T.; Abadie, T.; Matar, O. K.; Magri, L.. - In: PHYSICAL REVIEW FLUIDS. - ISSN 2469-990X. - 8:10(2023), pp. 1-21. [10.1103/PhysRevFluids.8.104302]

Availability:

This version is available at: 11583/2995077 since: 2024-12-07T19:36:56Z

Publisher:

American Physical Society - APS

Published

DOI:10.1103/PhysRevFluids.8.104302





Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

Data-driven modeling for drop size distributions

T. Traverso ^{1,2} T. Abadie ^{3,4} O. K. Matar ^{4,1} and L. Magri ^{2,1,*}

¹The Alan Turing Institute, British Library, London NW1 2DB, United Kingdom

²Department of Aeronautics, Imperial College London, South Kensington Campus,
London SW72AZ, United Kingdom

³School of Chemical Engineering, University of Birmingham, Birmingham B15 2TT, United Kingdom

⁴Department of Chemical Engineering, Imperial College London, South Kensington Campus,
London SW72AZ, United Kingdom



(Received 28 May 2023; accepted 3 October 2023; published 27 October 2023)

The prediction of the drop size distribution (DSD) resulting from liquid atomization is key to the optimization of multiphase flows from gas-turbine propulsion through agriculture to healthcare. Obtaining high-fidelity data of liquid atomization, either experimentally or numerically, is expensive, which makes the exploration of the design space difficult. First, to tackle these challenges, we propose a framework to predict the DSD of a liquid spray based on data as a function of the spray angle, the Reynolds number, and the Weber number. Second, to guide the design of liquid atomizers, the model accurately predicts the volume of fluid contained in drops of specific sizes while providing uncertainty estimation. To do so, we propose a Gaussian process regression (GPR) model, which infers the DSD and its uncertainty from the knowledge of its integrals and of its first moment, i.e., the mean drop diameter. Third, we deploy multiple GPR models to estimate these quantities at arbitrary points of the design space from data obtained from a large number of numerical simulations of a flat fan spray. The kernel used for reconstructing the DSD incorporates prior physical knowledge, which enables the prediction of sharply peaked and heavy-tailed distributions. Fourth, we compare our method with a benchmark approach, which estimates the DSD by interpolating the frequency polygon of the binned drops with a GPR. We show that our integral approach is significantly more accurate, especially in the tail of the distribution (i.e., large, rare drops), and it reduces the bias of the density estimator by up to 10 times. Finally, we discuss physical aspects of the model's predictions and interpret them against experimental results from the literature. This work opens opportunities for modeling drop size distribution in multiphase flows from data.

DOI: [10.1103/PhysRevFluids.8.104302](https://doi.org/10.1103/PhysRevFluids.8.104302)

I. INTRODUCTION

Liquid atomization is a phenomenon that appears in a large number of applications from agriculture [1] through drug delivery to fuel injection and cosmetics [2,3]. In most applications, it is desirable to have an *a priori* knowledge of the liquid dispersion structure, in particular, the distribution of droplet sizes as a function of the main control parameters and working conditions

*l.magri@imperial.ac.uk

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

of the atomizer. In agriculture, the droplet size and velocity distributions are important to ensure accurate delivery and retention of pesticide on the target [4] as the distribution of smaller drops determines the sensitivity of the coverage to environmental effects, such as wind [5]. A similar problem occurs in automotive paint sprays [6]. In respirable sprays for medicinal applications, the width of the drop size distribution (DSD) is of vital importance because drops smaller than a critical size are ejected from the body during exhalation, whereas droplets that are too large remain trapped in the respiratory system far upstream of their intended target [6]. In chemically reacting flows, before ignition, the DSD of atomized liquid fuels plays a central role in determining the thermal efficiency and the emissions of pollutants because it influences, e.g., the local evaporation rate of the fuel, the tendency of drops to be transported by the flow, and the spray penetration [7,8].

The dispersion structure of the atomized fluid is determined by a large number of factors, such as the presence of boundaries [9,10], the pressure drop across the nozzle or its shape [10], as well as the physical properties of the fluid itself, to name a few. For this reason, although the mathematical analysis of the governing equations describing sprays and jets can help understand the fundamental mechanisms leading to drop formation, it can provide only qualitative prediction regarding the DSD [11,12]. Thus, experimental investigation is often necessary to obtain quantitatively accurate information. Experiments provide high-fidelity data about the detailed shape of the DSD resulting from atomization; however, this is a low-throughput and expensive approach [13]. An alternative approach is to perform high-fidelity numerical simulations, which are a reliable tool to observe the details of the physical mechanisms at play in multiphase systems, e.g., Refs. [14–16]. However, high-fidelity simulations also have limitations because they are typically computationally expensive simulations, which make the systematic exploration of large design spaces difficult.

To mitigate the aforementioned challenges, Sacks *et al.* [17] modelled the deterministic output of numerical simulations as a stochastic process, providing the statistical basis for the optimal design of experiments (DOE), or active learning problem. In this framework, the goal is to choose the input sample carefully so that the information for the output quantity of interest (QoI) is maximized [18]. Recently, a similar framework was developed to design experiments using an output-weighted metric, with the objective of achieving the fastest possible convergence for the output statistics rather than reducing the accuracy in regions of the design space of little interest, with applications to rare events (e.g., waves) [19].

By modeling the output QoI of numerical simulations as a stochastic process it is possible to build a surrogate model (SM) of such QoI. In essence, a data-driven SM is a supervised learning algorithm that uses data to learn the mapping between points in the design (or input) space and the QoI. These SMs are sometimes referred to as *statistical emulators* as they *emulate* the solution of a physics-based model. The latter, which is numerically solved to obtain data, is thus called the *simulator* [20].

In the simplest case, the QoI is a scalar function of the input, e.g., the power output of a wind farm as a function of its layout and wind conditions [21]. However, if the QoI is the probability $p(d)$ of finding a drop with diameter d , then it becomes a function of the internal coordinate d . This intricate QoI cannot be easily modelled as a scalar stochastic process. To address this, a possible solution is to use multiple SMs, with each SM learning the function at a specific point of the internal coordinates. An illustrative example is provided in Ref. [20], where the QoI is represented by the time series of a scalar quantity of interest, specifically the carbon concentration within forests. Their approach involves training a vector-valued Gaussian process regressions (GPR) model, in which each element of the output vector represents the carbon concentration at a given time. Alternatively, the output vector could represent the coefficients of a set of basis functions, e.g., spectral decomposition [22,23], instead of the point evaluation of the QoI at different locations. Nonetheless, in all these cases the QoI is reconstructed from the knowledge of some *features*, such as its value at a set of points in time, or its spectral representation.

In this work we propose a SM to learn the mapping from the working conditions of a flat-fan liquid spray to the resulting DSD $p(d)$, which is the QoI. The working conditions are parametrized

by (i) the spray angle, (ii) the Reynolds, and (iii) the Weber numbers of the jet. To do so, we exploit the ability of GPR models to reconstruct the unknown function by using observations of any linear functional of it and not only of the point-evaluation functional. Other works exploited this idea in different contexts. For example, Longi *et al.* [24] used integral observations to optimize sensor placement, and Law *et al.* [25] to model the spatial distribution of malaria incidence from aggregate data, while Solak *et al.* [26] used derivative observations to model dynamical systems. In this work, we propose an algorithm that reconstructs the DSD by combining observations of integrals as well as of its first moment. These types of observations are relevant to multiphase systems, and especially to atomization problems, because the integrals of the DSD between two points d_1 and d_2 is directly related to the volume of atomized fluid contained in droplets with size $d_1 < d < d_2$. At the same time, by imposing the first moment of the DSD, we significantly reduce the bias of the estimator. The performance of such density estimators are compared to the case where pointwise evaluation of the DSD are used for its reconstruction, showing significant improvement. Finally, we remark how the proposed approach does not rely on *a priori* parametrization of the DSD in terms of known distributions (e.g., Gamma or log-normal), as done in the literature when reconstructing DSD from data, e.g., Refs. [12,27–29]. The prior knowledge on the typical shape of DSD is embedded in the choice of the kernel, which is a key aspect of the model selection.

The present work is centered on two main goals. First, we propose a methodology for constructing a data-driven SM for the prediction of DSDs in multiphase flows. Second, we produce high-fidelity data from numerical simulations of a flat-fan liquid spray to test the proposed methodology. This is performed by validating the SM’s predictions against the physical properties of the system. The paper is thus structured as follows. In Sec. II, we introduce the formalism of GPR adapted to include any observation that can be expressed as a linear functional of $p(d)$. Specifically, the functionals will be integrals of p , and their first moments. For the purpose of training and testing the SM, a data set of numerical simulations of the spray is produced at different working conditions. In Sec. III, we introduce the computational fluid dynamics framework used to perform these simulations and how the continuous DSD and its uncertainty are reconstructed from the populations of drops. In Sec. IV, the SM mapping the design parameters to the associated DSD is described, the performance is tested, and the outputs are interpreted from a physical perspective. Finally, concluding remarks are provided in Sec. V.

II. GAUSSIAN PROCESS REGRESSION WITH FUNCTIONAL OBSERVATIONS

Gaussian process models are a class of probabilistic statistical models, which can be used for supervised learning tasks, such as regression or classification, depending on the continuous or discrete nature of the output, respectively. Gaussian processes are principled alternatives to other machine learning models, e.g., artificial neural networks, Bayesian linear models, or support vector machines. A systematic and unified treatment of the topic of GPR models for supervised machine learning can be found in Williams and Rasmussen [30]. Numerous libraries to perform GPR are available in a number of open source and proprietary software, e.g., Python (such as GPy [31] and GPyTorch [32]) and Matlab (`fitgpr` function).

The goal of this section is to introduce the formalism of GPR to reconstruct an unknown function f from data. In doing so, we consider the general case in which data consists of observations of linear functionals $L[f]$ of the unknown function estimated by the Gaussian process, f . This extends the common case where L is the point-evaluation functional $L_x[f] = f(x)$, that is, when information about f consist of the knowledge of its value at discrete points. For example, in Sec. III, the unknown function $f = p(d)$ represents the DSD of a liquid spray, and the functionals used to reconstruct it are cumulative probabilities (i.e., integrals) and the first moment of f . This paper exploits both observations of the integrals and moments of a function for its reconstruction by means of GPR. The learning algorithm was implemented in an in-house code and is described in Sec. III B.

A. Functions as Gaussian processes

A Gaussian process (GP) is a collection of random variables that follow a joint Gaussian distribution. In the context of GPR, the random value represents the value of a function $f(\mathbf{x}) : \mathcal{X} \in \mathbb{R}^n \mapsto \mathbb{R}$ at point \mathbf{x} [30]. If we let $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ be any finite set of points in \mathcal{X} , then the vector $f(\mathbf{X}) = [f(\mathbf{x}_1), \dots, f(\mathbf{x}_N)]^T$ is modelled as a GP and it is denoted by

$$f(\mathbf{X}) \sim \mathcal{GP}(m(\mathbf{X}), k(\mathbf{X}, \mathbf{X})), \quad (1)$$

where $m(\mathbf{X})$ is the N -dimensional vector with elements $m(\mathbf{x}_i) = \mathbb{E}[f(\mathbf{x}_i)]$, i.e., the mean of f at \mathbf{x}_i , and $k(\mathbf{X}, \mathbf{X})$ is the N -by- N covariance matrix defined as

$$k_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \mathbb{E}[(f(\mathbf{x}_i) - m(\mathbf{x}_i))(f(\mathbf{x}_j) - m(\mathbf{x}_j))], \quad (2)$$

where $\mathbb{E}[y]$ is the expected value of the random variable y . The GP is defined by the knowledge of its mean and covariance functions.

Before the actual regression is performed (i.e., before data are considered), we chose the functions m and k based on prior knowledge about the unknown function f . In this work, such prior knowledge is embedded in the form of the kernel, $k(\mathbf{x}, \mathbf{x}'|\boldsymbol{\theta})$ (reported in Eqs. (24) and (27) below), which is chosen *a priori* and tuned by adjusting the hyperparameters $\boldsymbol{\theta}$. As more rigorously discussed in Secs. III B and IV A, these hyperparameters quantify how *fast* the function f changes as we move in the input space. This training is known as *type II maximum likelihood approximation* or ML-II [30]. The prior mean function is assumed to be zero, that is, $m(\mathbf{x}) = 0$ for all \mathbf{x} . This assumption preserves the generality of the method as it can always be enforced by shifting the data set by its mean.

Let us now consider a family of linear functionals of f , each denoted by $L_i[f] : \mathcal{F} \mapsto \mathbb{R}$ and such that $L_i[af + g] = aL_i[f] + L_i[g]$ for $f, g \in \mathcal{F}$, where \mathcal{F} is a reproducing kernel Hilbert space [33]. We consider that we have a noisy measurement of each L_i , denoted by $y_i = L_i[f] + \epsilon_i$, where ϵ_i is the zero-mean, additive, independent, and identically distributed Gaussian noise corrupting the observed value of L_i , and whose variance is denoted by $\mathbb{V}[\epsilon_i]$. As discussed in Sec. III B, these functional observations of f are definite integrals and the first moment of f . Here the unknown function is the DSD of a liquid spray, $f = p(d)$, and its observed values, y_i , will thus represent, e.g., the probability to find a droplet whose size is within a certain range (i.e., the extremes of integration defining L_i).

Considering the assumptions on ϵ_i , and that linear transformations of GPs are also GPs [34], the prior distribution for the i th observation y_i is also Gaussian:

$$y_i = \mathcal{N}(L_i[m], L_i^2[k] + \mathbb{V}[\epsilon_i]), \quad (3)$$

where $L_i^2[k]$ is expressed by

$$L_i^2[k] = L_i[L_i[k(\cdot, \mathbf{x}')]] = L_i[L_i[k(\mathbf{x}, \cdot)]], \quad (4)$$

where $y = \mathcal{N}(\mu, \sigma^2)$ indicates that the random variable y is normally distributed with mean μ and variance σ^2 . If the commonly used point-evaluation functional $L_i = L_{\mathbf{x}_i}[f] = f(\mathbf{x}_i)$ is employed, then Eq. (3) becomes $y_i = \mathcal{N}(m(\mathbf{x}_i), k(\mathbf{x}_i, \mathbf{x}_i) + \mathbb{V}[\epsilon_i])$ [30]. As Eq. (3) shows, a GP distribution on f gives a Gaussian distribution on any linear functional of f . This fact, together with the framework introduced in Secs. II B and II C, can be used to infer the integral of f from the knowledge of its value at a limited number of locations. This procedure, known as *Bayesian quadrature*, treats the problem of numerical integration as one of statistical inference [34] where the main source of uncertainty arises because it is not affordable to compute the function f at every location [35,36], e.g., when integrating functions over high-dimensional domain. In this work, we exploit this idea to perform the inverse task, that is, to infer f from the partial knowledge of some linear functionals of it (e.g., integrals).

B. Selecting a Gaussian process model: Training the hyperparameters

Let us consider the regression problem of estimating the value of $\mathbf{f}_* = f(\mathbf{X}_*)$ at some test locations $\mathbf{X}_* = \{\mathbf{x}_i\}_{i=1}^{N_*}$ from the knowledge of N_f features (i.e., linear functionals) of f . The definition of these functionals along with their observed values forms the data set $\mathcal{D} = \{(L_i, y_i)\}_{i=1}^{N_f}$. For brevity, we define the collection of all the functionals by $\mathbf{L} = \{L_i\}_{i=1}^{N_f}$. Each L_i is associated with its observed value, collected in the vector $\mathbf{y} = [y_1, \dots, y_{N_f}]^T$. This vector of observations is, in turn, associated with the noise vector, $\boldsymbol{\epsilon} = [\epsilon_1, \dots, \epsilon_{N_f}]^T$, and with the N_f -by- N_f diagonal matrix containing the variance of such noise along the main diagonal (i.e., uncorrelated noise), $\mathbb{V}[\boldsymbol{\epsilon}]$.

The GPR comprises two main steps in which the data set \mathcal{D} is used to (i) select a model, and (ii) condition the model on the observations, respectively. The model selection consists of optimizing the hyperparameters $\boldsymbol{\theta}$ defining the covariance function $k = k(\mathbf{x}, \mathbf{x}'|\boldsymbol{\theta})$ based on the observed data, which is the *training*. This is achieved by maximizing the marginal likelihood $\mathcal{L}(\boldsymbol{\theta}) = p(\mathbf{y}|\mathbf{L}, \boldsymbol{\theta})$ given by [30]

$$\log p(\mathbf{y}|\mathbf{L}, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{y}^T k_y^{-1} \mathbf{y} - \frac{1}{2} \log |k_y| - \frac{N_f}{2} \log 2\pi, \quad (5)$$

where $k_y = k_L(\mathbf{L}, \mathbf{L}) + \mathbb{V}[\boldsymbol{\epsilon}]$ is the covariance between the noisy observations of \mathbf{L} , and where the N_f -by- N_f covariance matrix of the noise-free observations, $k_L(\mathbf{L}, \mathbf{L}) = k_{L,ij}$, is obtained by exploiting the bilinearity of covariance [37], yielding

$$k_{L,ij} = L_i[L_j[k(\cdot, \cdot)]] = L_j[L_i[k(\cdot, \cdot)]]. \quad (6)$$

On the right-hand side of Eq. (5), the first term, $-\frac{1}{2}\mathbf{y}^T k_y^{-1} \mathbf{y}$, involves the value of the observations \mathbf{y} and is proportional to the data fit; the second term, $\frac{1}{2} \log |k_y|$, can be interpreted as a penalty parameter that promotes smoother, slowly varying, thus, simpler functions; and the last term is a normalization constant. The definition of the functionals used in this work and the associated covariance terms in Eqs. (6) are explicitly reported in Sec. III B. Once the model is trained, we denote the optimized covariance function by $\hat{k} = k(\mathbf{x}, \mathbf{x}'|\hat{\boldsymbol{\theta}})$, where $\hat{\boldsymbol{\theta}}$ corresponds to

$$\hat{\boldsymbol{\theta}} \equiv \arg \max_{\boldsymbol{\theta}} p(\mathbf{y}|\mathbf{L}, \boldsymbol{\theta}), \quad (7)$$

and similarly for the covariance matrix between observations of different functionals, $\hat{k}_{L,ij} = L_i[L_j[\hat{k}(\cdot, \cdot)]]$.

C. Posterior distribution

Once the model is trained, the next step is to condition the prior, see Eq. (1), on the data, \mathcal{D} . In practice, that is to find the posterior distribution for (i.e., prediction of) the value of f at the test locations, \mathbf{f}_* . These predictive equations for \mathbf{f}_* [30] are here generalized to observations of functionals other than the point-evaluation functional. They read

$$\mathbf{f}_*|\mathcal{D}, \mathbf{X}_* \sim \mathcal{N}(\mathbb{E}[\mathbf{f}_*], \text{cov}(\mathbf{f}_*)), \quad \text{where} \quad (8)$$

$$\mathbb{E}[\mathbf{f}_*] = \hat{k}_L(\mathbf{L}_*, \mathbf{L}) \hat{k}_y^{-1} \mathbf{y}, \quad (9)$$

$$\text{cov}(\mathbf{f}_*) = \hat{k}_L(\mathbf{L}_*, \mathbf{L}_*) - \hat{k}_L(\mathbf{L}_*, \mathbf{L}) \hat{k}_y^{-1} \hat{k}_L(\mathbf{L}, \mathbf{L}_*), \quad (10)$$

where $\hat{k}_y = \hat{k}_L(\mathbf{L}, \mathbf{L}) + \mathbb{V}[\boldsymbol{\epsilon}]$ and where $\mathbf{L}_* = \{L_x[f]|\mathbf{x} \in \mathbf{X}_*\}$ denotes the set of point-evaluation functionals at the test locations \mathbf{X}_* . In other words, in Eq. (9) we now use the optimized covariance (\hat{k}_y) and the data (\mathbf{y}) to predict the most likely value of the unknown function f at a set of test locations (\mathbf{X}_*) in which we have no data, i.e., we compute $\mathbb{E}[\mathbf{f}_*] = \mathbb{E}[f(\mathbf{X}_*)]$. Equation (10) is employed to compute the covariance between these predictions, $\text{cov}(\mathbf{f}_*)$. Its diagonal represents the variance of the prediction at each test point, \mathbf{X}_* , thus quantifying its accuracy, and it is independent

of the value of the observations, \mathbf{y} . As discussed in Sec. IV, this fact is exploited to optimally design part of the numerical experiments because it is possible to know how a new experiment affects the uncertainty of the prediction before it is run.

In Secs. III B and III C we use the predictive Eqs. (8)–(10) to predict the DSD of a liquid spray, $f = p(d)$, that is, the probability of finding a spherical drop with diameter d . We do so by using a specific form of the covariance function that well encodes our prior knowledge about the general characteristics of DSD (e.g., heavy tail distributions). Finally, in Sec. IV, we model each of these features of p using as many GPR models, effectively treating them as unknown functions of the working conditions of the spray, which is the input space of the SM. By doing so, we obtain a SM that maps points in the space of the working conditions of the sprays to the associated DSD and its uncertainty.

III. GPR FOR PROBABILITY DENSITY ESTIMATION FOR SPRAYS

In this section, we propose a novel application of the framework explained in Sec. II in which observations include (i) definite integrals and (ii) the first moments of the unknown function. The unknown function represents the unobservable underlying probability density (PDF) that generates a finite data sample, that is, the data. Each element of the sample, or population, corresponds to the diameter of a drop within a liquid spray. To obtain the population of drops, a flat-fan liquid spray is modelled by means of numerical simulations, as described in Sec. III A. In Sec. III B, we then discuss how the GPR framework is used to estimate the underlying PDF generating the DSD. Finally, the results are discussed in Sec. III C.

A. Numerical simulations of a liquid jet

1. Governing equations and dimensionless parameters

Atomization is the physical process by which an initial liquid, here a liquid sheet, is fragmented into smaller droplets due to a cascade of events. In the initial phase, the sheet's free surface is linearly unstable, which means that perturbations grow exponentially until nonlinear effects cause the formation of ligaments. In a second phase, ligaments breakup into a number of smaller drops, which are responsible for the shape of the DSD of the spray [12]. These phases are referred to as *primary* and *secondary* atomization, respectively. The dynamics of an incompressible, Newtonian liquid jet are described by the one-fluid formulation of the incompressible Navier-Stokes equations, which under the assumptions of isothermal flow, and in the absence of interfacial mass transfer, are expressed by

$$\nabla \cdot \mathbf{u} = 0, \quad (11)$$

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot [\eta(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \mathbf{F}_\sigma + \mathbf{g}, \quad (12)$$

where ρ , η , and \mathbf{u} are the local density, dynamic viscosity, and velocity, respectively; \mathbf{g} is the acceleration due to gravity, and the \mathbf{F}_σ is the capillary force, which can be written as $\mathbf{F}_\sigma = \sigma \kappa \mathbf{n} \delta_s$, with σ , \mathbf{n} , κ , and δ_s denoting the surface tension (considered constant), the unit normal to the interface, the local curvature of the interface, and the Dirac delta distribution associated with the interface, respectively. We consider an ellipsoidal nozzle with minor and major semiaxis r_a and r_b , respectively. Using twice the nozzle minor semiaxes R_0 , the jet maximum velocity U_0 , $t_c = R_0/U_0$, and $\rho_l U_0^2$ as the characteristic length, time, velocity, and pressure, respectively, and the liquid density ρ_l and viscosity η_l to scale the density and viscosity, respectively, the equations above can be written as [38,39]

$$\tilde{\nabla} \cdot \tilde{\mathbf{u}} = 0, \quad (13)$$

$$\tilde{\rho}(\partial \tilde{\mathbf{u}} \partial \tilde{t} + \tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{\mathbf{u}}) = -\tilde{\nabla} \tilde{p} + 1 \text{Re} \tilde{\nabla} \cdot [\tilde{\eta}(\tilde{\nabla} \tilde{\mathbf{u}} + \tilde{\nabla} \tilde{\mathbf{u}}^T)] + 1 \text{We} \tilde{\kappa} \tilde{\delta}_s \mathbf{n} + 1 \text{Bo} \tilde{\mathbf{g}}, \quad (14)$$

where the tilde denotes dimensionless quantities. Considering flows where the capillary forces dominate the acceleration due to gravity ($\text{Bo} = \rho_l g R_0^2 / \sigma \ll 1$), the two dimensionless parameters that influence the spray dynamics, thus the resulting droplet size distributions, are the Reynolds number $\text{Re} = \rho_l U_0 R_0 / \eta_l$ and the Weber number $\text{We} = \rho_l U_0^2 R_0 / \sigma$.

2. Numerical methods

We perform numerical simulations of a liquid jet using the *Basilisk* GPL-licensed computational code [40]. The Navier-Stokes equations are solved using the projection method [41], while the volume-of-fluid method is used to capture and reconstruct the interface [42,43]. Within this framework, the volume fraction C , which equals unity in the liquid phase and zero in the ambient phase, is passively advected with the flow:

$$\frac{\partial C}{\partial \tilde{t}} + \tilde{\mathbf{u}} \cdot \nabla C = 0. \quad (15)$$

The local fluid viscosity η and density ρ are obtained by linearly averaging the liquid and gas properties with C ,

$$\tilde{\eta} = \frac{\eta_l C + (1 - C)\eta_g}{\eta_l}, \quad \tilde{\rho} = \frac{\rho_l C + (1 - C)\rho_g}{\rho_l}, \quad (16)$$

where ρ_g and η_g represent the gas density and viscosity, respectively, and $\rho_l / \rho_g = \eta_l / \eta_g = 10$ throughout. Surface tension forces are computed from the local normal and curvature of the interface, which are estimated from C using the height-functions method [41,44].

The simulations are performed on a cube with nondimensional linear size $\tilde{L} = 88$. On each side, we impose reflective boundary conditions (i.e., zero normal derivative) for all the state variables exceptions made for the pressure on the opposite side of the nozzle, which is set to zero. The inflow velocity, $\tilde{\mathbf{u}}_{\text{in}} = [\tilde{u}_{x,\text{in}}(z), 0, \tilde{u}_{z,\text{in}}(z)]$, is imposed on an ellipsoidal region on the plane $x = 0$ with major axes r_b along direction z and minor axes r_a along direction y (aspect ratio $r_b/r_a = 6$), see Fig. 1(a). The components of $\tilde{\mathbf{u}}_{\text{in}}$ are plotted in Fig. 1(b); they are defined such that at each point inside the ellipse

$$\sqrt{\tilde{u}_{x,\text{in}}^2 + \tilde{u}_{z,\text{in}}^2} = U_0 \text{erf} \left[\frac{2(r_b - |z|)}{\delta} \right] \\ \tilde{u}_{z,\text{in}} / \tilde{u}_{x,\text{in}} = z \tan \alpha / r_b, \quad (17)$$

which ensures that the desired spray angle α is obtained, and $\delta = r_b/4$.

The solver employs local adaptive mesh refinement [45] using the octree mesh type. Basilisk's procedure for local mesh adaptation employs the wavelet transform to assess the discretization error of a given scalar field. In our simulations, we adapt the mesh according to the error on the local nondimensional fluid velocity \mathbf{u} , and the gradient of the volume fraction of the fluid in the jet ∇C , which indicates the vicinity of an interface. Throughout this study, the threshold error value is set to 0.1, and the maximum level of refinement is set equal to 11. Figures 1(c) to 1(f) show four jets discretized by $O(10^7)$ grid points corresponding with different working conditions of the jet, i.e., different values of α , Re and We (see caption). The simulations are concluded when a sufficient number of droplets has formed, typically $O(10^3)$, and can take up to 72 hours on 24 cores Intel i7-12700.

When the simulation is concluded only droplets with an equivalent diameter d_i such that $d_{\min} < d_i < d_{\max}$ with $[d_{\min}, d_{\max}] = [0.08, 0.8]$ are considered (minimal cell size is $l_{\min} \approx 0.04$). Then nonspherical drops are excluded from the sample by applying a high-pass filter based on their sphericity $\phi_i = \pi^{1/3} (6V_i)^{2/3} / A_i$. To estimate the surface area of the i th drop, A_i , we first note that the vector field ∇C approximates a Dirac delta function on the interface between the drop and the ambient fluid. Within a computational cell on such an interface, we define the vector $d\mathbf{A}$ whose modulus is equal to the surface area and its orientation is perpendicular to it. The area of such an

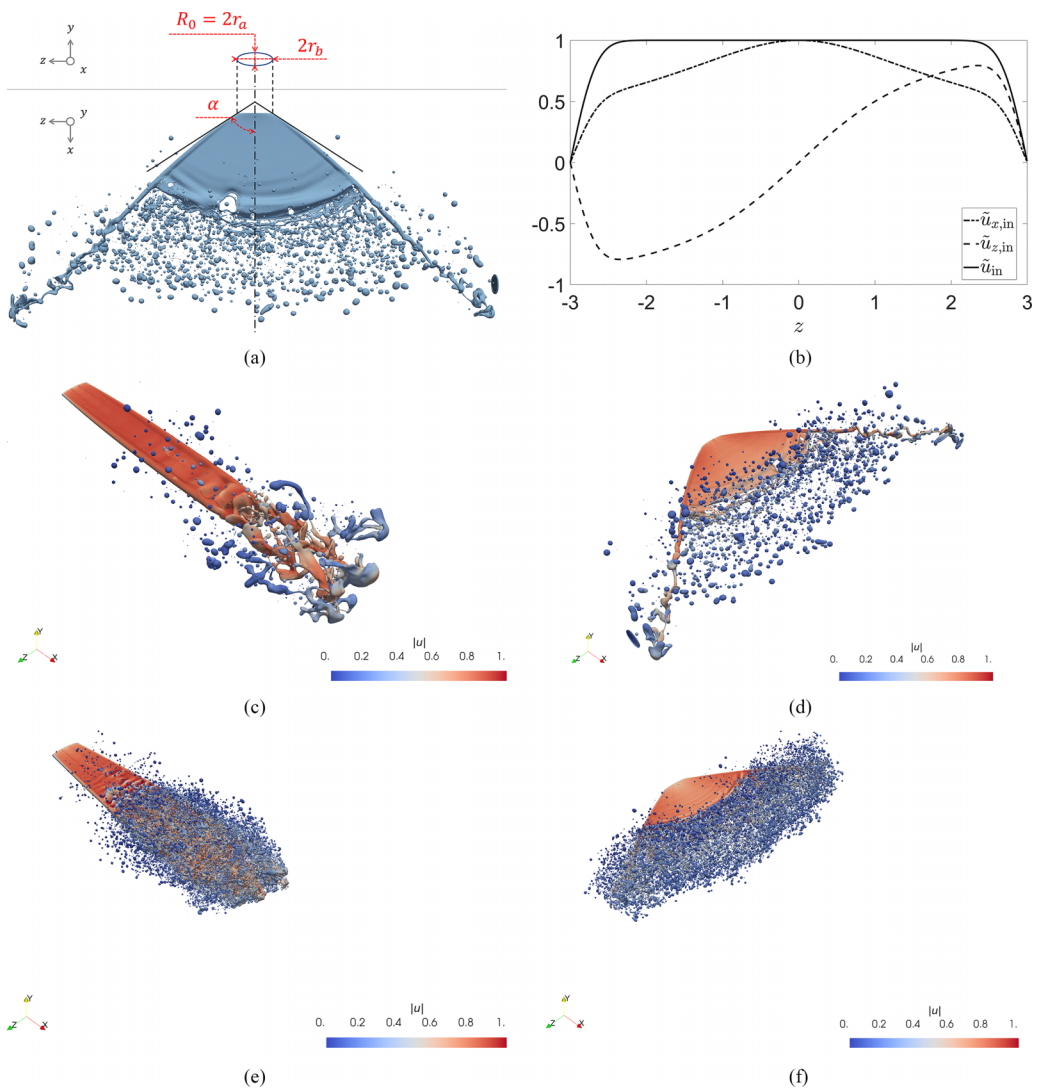


FIG. 1. (a) Qualitative representation of the spray angle and the ellipsoidal region (through orthographic projection) letting the inflow into the domain. (b) Axial ($\tilde{u}_{x,in}$), lateral ($\tilde{u}_{z,in}$), and total inflow velocity, $\tilde{\mathbf{u}}_{in} = (\tilde{u}_{x,in}(z), 0, \tilde{u}_{z,in}(z))$, evaluated along the main diagonal of the ellipse, i.e., $x = 0$, $y = 0$, and $z \in [-r_b/2, r_b/2]$ with $r_b = 6$. Values relative to a spray angle of 60° , which corresponds to $\tan^{-1}(\tilde{u}_{z,in}/\tilde{u}_{x,in})$ as $z \rightarrow 0$. Fluid interface and velocity magnitude (color) of four sprays with (c) $Re = 20$, $We = 18$, $\alpha = 10^\circ$; (d) $Re = 20$, $We = 18$, $\alpha = 65^\circ$; (e) $Re = 59$, $We = 90$, $\alpha = 10^\circ$; and (f) $Re = 59$, $We = 90$, $\alpha = 65^\circ$.

interface is then given by $|d\mathbf{A}| = (dA_1^2 + dA_2^2 + dA_3^2)^{1/2}$, where dA_j ($j = 1, 2, 3$) are expressed by

$$dA_j = \int_{V_{cell}} \nabla C_j dV. \quad (18)$$

Finally, A_i is computed by summing all the infinitesimal contributions dA_j relative to the i th drop. Drops with $\phi_i < 0.9$ are thus excluded as they are likely to be ligaments that would eventually break up into smaller drops, thus corrupting the statistics of the DSD. We observe that, on average, around 3% of the total number of drops are filtered out based on sphericity, corresponding to 11% of the

total atomized volume. If the filter based on the size is also considered, then the drops included in the final count are on average 74% of the total, making up for 46% of the total volume. Analogous size-based band-pass and sphericity-based filters are applied also when using optical techniques to study atomization due to the intrinsic limits of the image acquisition system [46–48]. In the range of parameters explored in this paper, and discussed in Sec. IV, we observe that the results remain qualitatively unchanged when we remove the sphericity filter and the values of d_{\min} and d_{\max} are varied. Specifically, when the filter on the sphericity is not applied, the tails of the DSDs are slightly heavier for all the spray conditions tested (result not shown). This follows from the fact that larger drops are more likely to be nonspherical due to the reduced importance of surface tension compared to small drops.

B. Features of the drop size distribution and their covariance

1. Observations of the mean and cumulative probabilities

The drop population obtained from the numerical simulation of the spray is treated as an independent and identically distributed sample of N_d events, $\mathcal{S} \equiv \{d_i\}_{i=1}^{N_d}$. The unknown distribution from which the sample is drawn is denoted by $p(d)$ defined on the interval between the smallest and the largest measurable diameters, $[d_{\min}, d_{\max}]$. Our goal is to use GPR to infer $p(d)$ and its uncertainty from \mathcal{S} . The first step is to produce a frequency plot of the data with a suitable binning scheme, \mathcal{B} , defined by the bins' edges b_i such that $\mathcal{B} = \{b_i | b_1 = d_{\min}, b_{N_b+1} = d_{\max}, b_{i+1} > b_i\}_{i=1}^{N_b+1}$. Let N_b be the number of bins, h_i the width of the i th bin, and n_i the number of drops within the i th bin. The cumulative probability of a drop being in the first i bins is approximated as

$$P_i \equiv P(b_{i+1}) = \int_{d_{\min}}^{b_{i+1}} p(x) dx \approx \frac{\sum_{j=1}^i n_j}{N_d (b_{i+1} - d_{\min})} \quad \text{for } i = 1, \dots, N_b. \quad (19)$$

Hence, we compute the mean of the population, \bar{d} , which is an unbiased estimator of the first moment of the distribution μ_1 , yielding

$$\mu_1 \equiv \int_{d_{\min}}^{d_{\max}} p(x) x dx \approx \bar{d} = \frac{\sum_{i=1}^{N_d} d_i}{N_d}. \quad (20)$$

The quantities P_i and μ_1 are linear functionals of $f(x)$, which we denote $L_i[p] \equiv P_i$, and $L_{N_f}[p] \equiv \mu_1$. The data set \mathcal{D} is thus represented by (i) the definitions of these functionals, (ii) their observed values $\mathbf{y} = [P_1, \dots, P_{N_b}, \mu_1]$, and (iii) the variance of the noise corrupting these observations, $\mathbb{V}[\epsilon]$.

This data set is used to estimate the density p at the test locations $\mathbf{X}_* = [d_1, \dots, d_{N_*}]$ with the predictive Eqs. (8) to (10), where $\mathbf{f}_* = p(\mathbf{X}_*)$. To do so, the covariance between different measured features of p is computed with Eq. (6). Specifically, the covariance between the observation of the i th cumulative probability, P_i , and that of the first moment, μ_1 , is

$$k_L(P_i, \mu_1) \equiv L_i[L_{N_f}[k(\cdot, \cdot)]] = L_{N_f}[L_i[k(\cdot, \cdot)]] = \int_{d_{\min}}^{d_{\max}} \int_{d_{\min}}^{b_{i+1}} k(z, z') z dz dz'. \quad (21)$$

Likewise, the covariance between the observations of P_i and P_j is

$$k_L(P_i, P_j) \equiv L_i[L_j[k(\cdot, \cdot)]] = L_j[L_i[k(\cdot, \cdot)]] = \int_{d_{\min}}^{b_{j+1}} \int_{d_{\min}}^{b_{i+1}} k(z, z') dz dz'. \quad (22)$$

Finally, the covariance between the observations of P_i and the function value at the j th test point, $p(d_j) = L_{d_j}$, is

$$k_L(P_i, L_{d_j}) \equiv L_i[L_{d_j}[k(\cdot, \cdot)]] = L_{d_j}[L_i[k(\cdot, \cdot)]] = \int_{d_{\min}}^{b_{i+1}} k(z, d_j) dz = \int_{d_{\min}}^{b_{i+1}} k(d_j, z) dz. \quad (23)$$

2. Nonstationary covariance function

We choose the specific form of the kernel $k = k(d, d')$ to compute the covariance in Eqs. (21), (22), and (23) guided by the prior knowledge about the function we are learning. A number of studies have shown that the DSDs emerging from the breakup of ligaments and drop impacts can be approximated by Gamma [49,50] as well as log-normal [51] distributions. If the function $p(d)$ has similar shape to these distributions, then we expect rapid variation of the density p with the input d around the peak, followed by a progressive transition to slower variation of the density towards the tail of the distribution (i.e., at large diameters). Consequently, the typical distance $\Delta d = |d - d'|$ over which the correlation between $p(d)$ and $p(d + \Delta d)$ decays increases with the input, i.e., Δd increases towards the tail of the distribution. A covariance function $k = k(d, d')$ with this property is said to be *nonstationary*, as opposed to *stationary* covariances, for which the correlation decays everywhere with the same law, $k(d, d') = k(\Delta d)$. We allow the GPR model to capture this property of DSDs by employing the nonstationary version of the squared exponential covariance function proposed by [52]

$$k(d, d') = \sigma_f^2 \left[\frac{2l_d(d)l_d(d')}{l_d^2(d) + l_d^2(d')} \right]^{1/2} \exp \left[-\frac{(d - d')^2}{l_d^2(d) + l_d^2(d')} \right]. \quad (24)$$

The covariance in Eq. (24) is constructed so that the positive function $l_d(d)$ can be interpreted as the local correlation length, which is a function of the input d . Alternatively, for a given value of the parameter σ_f , the value of l_d can be interpreted as the inverse of the characteristic gradient of p .¹ Small values of l_d imply that, locally, the modelled function varies significantly due to small variation of the input. This happens near the peak, where the model has high capacity, that is, high ability to fit irregular data. Similarly, large values of l_d imply that significant variations of the output require large variation of the input, i.e., in the tail the model has low capacity. Finally, we note that the covariance function in Eq. (24) simplifies to the stationary squared exponential kernel if we let l_d be constant, corresponding to a model that has uniform capacity with the input. Given the sharply peaked shape of DSD emerging from liquid atomization, a model with a stationary kernel (such as the squared exponential kernel) would either underfit the peak or overfit the tail of the distribution (results not shown).

In the following, we consider two possible forms for l_d : A linear form, as it is the simpler form of nonuniformity, and the logarithmic form, to reproduce the nonlinear behavior of log-normal and Gamma distributions, respectively

$$l_{\text{lin}}(d) = a_0 + a_1 d, \quad \text{and} \quad l_{\text{ln}}(d) = b_0 + b_1 \ln(b_2 d). \quad (25)$$

Consequently, the hyperparameters characterizing the kernel are either $\theta \equiv \theta_{\text{lin}} = \{\sigma_f, a_0, a_1\}$ or $\theta \equiv \theta_{\text{ln}} = \{\sigma_f, b_0, b_1, b_2\}$, depending on whether we use $l_{\text{lin}}(d)$ or $l_{\text{ln}}(d)$, respectively.

The noise that affects each measurement is assumed to be known and, in general, $\mathbb{V}[\epsilon_i] \neq \mathbb{V}[\epsilon_j]$ for $i \neq j$. This scenario is relevant to real-world engineering and industrial applications, in which the accuracy of the experimental apparatus is a known function of the measured value, estimated during their calibration. In other words, we assume that the data points are obtained in a controlled environment with reliable estimates of their uncertainties. As a result, it is not necessary to either (i) treat the noise variance of each observation as an additional hyperparameter to be optimized during training (*heteroscedasticity*), which would significantly increase the number of trained parameters, and (ii) introduce the assumption that the variance is the same for all observations (*homoscedasticity*). The latter assumption would be unrealistic in the present context due to the

¹As pointed out by Gibbs [52] (chapter 3.10.2), who proposed the correlation function, this interpretation is not applicable for any arbitrary form of $l_d(d)$ due to the effect of the prefactor to the exponential on the right-hand side of Eq. (24). However, for the form of $l_c(d)$ used in this work, i.e., monotonic functions, see Eq. (25), this interpretation is accurate.

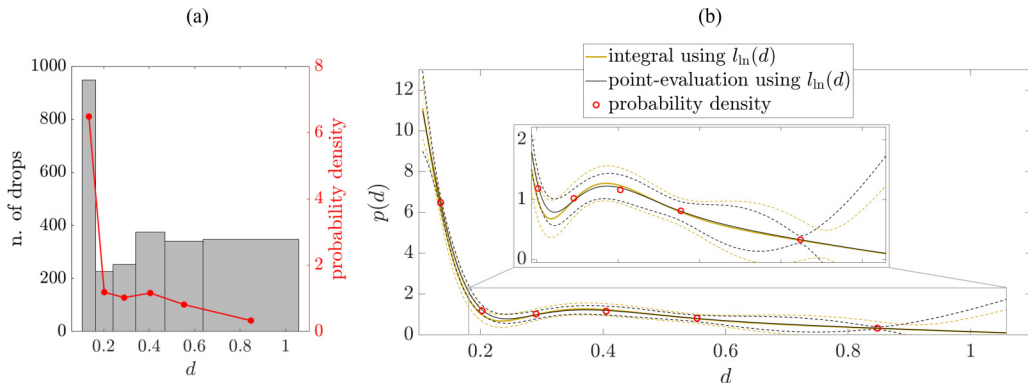


FIG. 2. Data from the numerical simulation with $\alpha = 65^\circ$, $Re = 20$, and $We = 18$, see also Fig. 1(d). (a) Histogram of the drops count in the simulated spray using six bins and associated probability (red circles). (b) DSD predicted by the GPR using observations of the mean and cumulative probabilities p_{int} (solid black lines) and pointlike observations p_{pt} (dashed gray lines). The dashed lines represent the 95% confidence intervals. The linear length scale $l_{lin}(d)$ is employed to perform both regressions.

difficulties of measuring drops of very different sizes with the same accuracy, whether employing numerical or experimental techniques.

The optimization problem in Eq. (7) for the training of the GPR is solved with a genetic algorithm [53]. Even though a gradient-based approach would be computationally more efficient, our choice is justified by its simplicity of implementation and by the small computational load of the present problem, in which the number of data points used to perform the regression (i.e., the number of bins) is $O(10)$ for each DSD. The integrals in Eqs. (21), (22), and (23) are computed with a second-order finite-difference scheme by discretizing k into $O(10^2)$ points. Each regression, such as those reported in the following, takes between $O(10^1)$ and $O(10^2)$ seconds to complete on an eight-core Intel i9 12900 processor.

C. Estimation of drop size distribution

The GPR framework discussed in Sec. III B is employed to estimate the DSD of a liquid spray. To evaluate the performance of the method, we refer to data relative to a spray characterized by $\alpha = 65^\circ$, $Re = 20$, and $We = 18$ (see Fig. 1). On filtering based on sphericity and size, the drops are sorted into bins [Fig. 2(a)]. The corresponding probability density (red circles) is computed as $\pi_i \equiv n_i / (N_d h_i)$. The area under this line, up to each bin's edge, is equal to the cumulative probability P_i , defined in Eq. (19). The thick golden line in Fig. 2(b) shows the posterior mean of $p_{int}(d)$ inferred by the GPR by imposing the integrals P_i and the value of the mean, μ_1 , computed according to Eq. (20) (dashed lines are 95% confidence intervals). Typically, the measurement accuracy is a function of the magnitude of the signal [54]. To mimic this behavior, in this example the variance of the additive Gaussian noise is set equal to 5% of the measured value of each observation, that is, $\mathbb{V}[\epsilon_i] = y_i \times 0.05$, except for the first two bins (P_1 and P_2), for which the error is assumed to be double. This reflects the fact that smaller drops are modelled less accurately, as their size approaches the spatial resolution of the solver.

1. Comparison between integral and pointlike observations

An alternative method for inferring the DSD using GPR is to use traditional point-evaluation functionals as observations. These are the red circles corresponding to the probability density in Fig. 2(a), which are treated as observations of the DSD at the center of each bin (i.e., the equivalent of interpolating the edges of a *frequency polygon* [55] with a GP model). By doing so, and by using

TABLE I. Data from the numerical simulation with $\alpha = 65^\circ$, $\text{Re} = 20$, and $\text{We} = 18$; drops sorted in six bins, and $l_{\text{in}}(d)$; see also Fig. 2. Relative percentage error in the probability associated with each bin using integral observations, $e_{i,\text{int}}$, and using point-evaluation observations, $e_{i,\text{pt}}$. The last line reports the relative percentage error of the mean of the estimator.

Bin	$e_{i,\text{int}}(\%)$	$e_{i,\text{pt}}(\%)$
1	3.23	6.84
2	7.66	28.53
3	2.63	2.15
4	5.39	2.41
5	0.08	1.31
6	0.63	3.25
e_μ	0.27	3.0

the same covariance function and observations' noise variance, one obtains $p_{\text{pt}}(d)$ [thin black curve in Fig. 2(b)].

The two methods give qualitatively similar results; however, some important quantitative differences need be pointed out. First, we define the relative error between the population mean and the first moment of the reconstructed DSD, $e_\mu = 100|\bar{d} - \mu_1|/\bar{d}$. As reported in Table I, the expected value of p_{int} deviates by the mean diameter \bar{d} by 0.27%, while that of p_{pt} by 3.0%, corresponding to a 10-fold reduction in the bias of the estimator p_{int} . Second, the probability that a drop with diameter $b_i < d < b_{i+1}$ forms, i.e., the integrals

$$p_i \equiv \int_{b_i}^{b_{i+1}} p(x) dx, \quad (26)$$

are compared with the associated quantity computed from the population of drops, that is, π_i . The error in the prediction is quantified by $e_{i,\text{int}}$ and $e_{i,\text{pt}}$ computed as $e_i = 100|p_i - \pi_i|/\pi_i$, where $p = p_{\text{int}}$ and $p = p_{\text{pt}}$ in Eq. (26) are used, respectively. The values of $e_{i,\text{int}}$ and $e_{i,\text{pt}}$ are listed in Table I. We observe that the predictions provided by p_{int} are more accurate for all the bins except 3 and 4 and that the error is reduced by up to two orders of magnitude in the tail of the distribution. The value of p_i plays an important role as it is a direct indication of the volume (thus mass) of atomized fluid contained in drops with diameter $b_i < d < b_{i+1}$. As further discussed in Sec. V, this property of the DSD is key to practical applications.

2. Effect of the kernel and binning scheme

The choice of the kernel and the binning scheme are key steps in model selection as they affect the prediction. In this section, we discuss how the proposed approach allows us to mitigate the effect of these choices on the predicted DSD. The kernel defines the GPR model structure and encodes our assumptions about the functions that we are learning, which makes it a crucial ingredient in a GPR model [30]. Although it is clear from the context that our covariance function should be nonstationary, as discussed in Sec. III B 2, it is not obvious which specific form the kernel should have. We compare the predicted DSD obtained using the linear ($p_{\text{int,lin}}$) and the logarithmic ($p_{\text{int,ln}}$) length scale in Fig. 3(b). The two curves are obtained from the same observations as in Fig. 2. The optimized lengths scales relative to p_{int} and $p_{\text{int,ln}}$ are also plotted in Fig. 3(a). By comparing the two curves in Fig. 3(b), the choice of a linear form of the length scale removes the presence of the peak at small d , typical of log-normal and Gamma distributions, which are characterized by nonlinear decaying first derivatives. By design, the proposed method ensures that the bias of the estimator of the DSD (quantified by e_μ) and the cumulative probabilities have similar accuracy in both cases, as shown by the comparison of the values of $e_{i,\text{int}}$ in Table I and Table II. The fact that cumulative

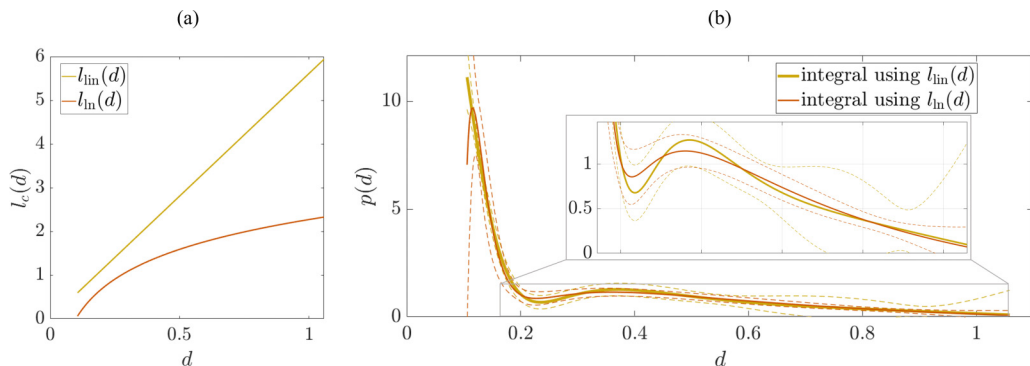


FIG. 3. Data from the numerical simulation with $\alpha = 65^\circ$, $\text{Re} = 20$, and $\text{We} = 18$; drops sorted in six bins as shown in Fig. 2(a). (a) Optimized characteristic length scales l_{in} and l_{n} , corresponding to the GPR models in Figs. 2(b) and 3(b), respectively. (b) DSD predicted by the GPR model using observations of the mean and cumulative probabilities $p_{\text{int}}(d)$ employing the logarithmic length scale $l_{\text{n}}(d)$.

probabilities and bias of the DSD are robustly predicted against the choice of the kernel is a key feature of the proposed model.

When performing high-fidelity numerical simulations of atomization processes, e.g., Ref. [15], small drops require high resolution to be resolved, and large drops are rare events, thus requiring the simulation to run for a long time to obtain a sufficient amount of realizations and for the statistics to converge. For this reason, obtaining large samples of drops is time consuming, thus the higher the fidelity of the simulation, the fewer the number of drops observed, and the coarser the binning scheme. We mimic this scenario in which fewer drops are observed by sorting the same population of drops in the examples above using a coarse binning scheme with only three bins. We train two models, one with integral and one with pointwise observations, using the resulting histogram reported in Fig. 4(a). The resulting DSDs predicted by the models are shown in Fig. 4(b). By comparing these DSDs with the red curve in Fig. 3(b) (obtained using the same kernel but more bins), we note that the details of the corresponding DSDs are different from those obtained using the more refined binning scheme. However, by inspection of Table II, we observe that the magnitude of all the errors, e_i and e_μ , remains of the same order of magnitude (see Table I) when integral observations are used. Finally, Fig. 4(b) shows that, when point-evaluation observations are used, the density estimator (i) is biased (last column of Table II), (ii) fails to capture the magnitude of the peak, and (iii) overestimates the probability in the tail.

TABLE II. Relative percentage error in the probability associated with each bin, $e_{i,\text{int}}$ and $e_{i,\text{pt}}$, relative to the DSDs in Fig. 4(b) with $l_{\text{n}}(d)$. The last line reports the relative percentage error of the mean of the estimator.

Bin	$e_{i,\text{int}}(\%)$	$e_{i,\text{pt}}(\%)$
1	0.67	13.20
2	2.68	9.31
3	1.30	9.81
e_μ	0.57	6.27

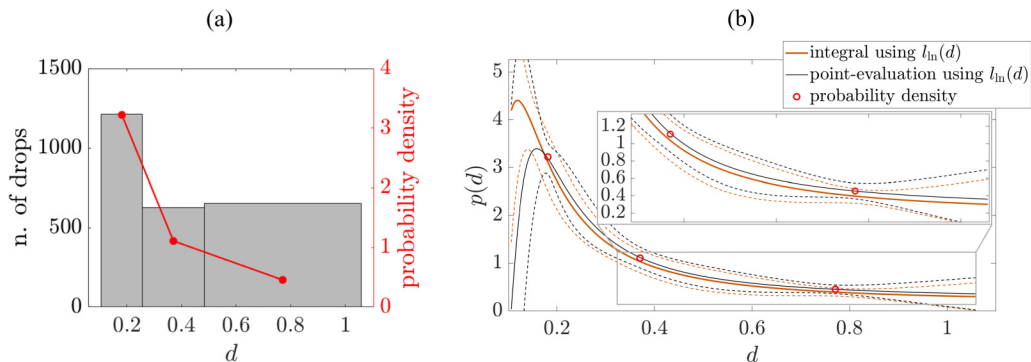


FIG. 4. Data from the numerical simulation with $\alpha = 65^\circ$, $Re = 20$, and $We = 18$. (a) Histogram of the drops count in the simulated spray using three bins and associated probability (red circles). (b) DSD predicted by the GPR using observations of the mean and cumulative probabilities p_{int} (solid green lines) and pointlike observations p_{pt} (dashed gray lines). The dashed lines represent the 95% confidence intervals. The logarithmic length scale $l_{log}(d)$ is employed in both models.

IV. GPR-BASED SURROGATE MODEL OF A LIQUID SPRAY

In this section, we use the method for the reconstruction of the DSD introduced in Sec. II and tested in Sec. III to develop a data-driven SM of a flat fan spray. The structure of the SM is introduced in Sec. IV A, and the predictions are discussed in Sec. IV B.

A. Structure and training of the surrogate model

The SM is designed to predict the DSD of the spray, $p(d|\mathbf{x})$, given the control/design parameters $\mathbf{x} = [\alpha, Re, We]$. This comprises two main steps, as depicted in Fig. 5. First, in Step 1, the SM learns how the features of the DSD, $\mathbf{y} = [y_1, \dots, y_{N_f}]$, change with \mathbf{x} . Then, their predicted value at a test location, $\mathbf{y}(\mathbf{x}_*)$, and the associated uncertainty, $\mathbb{V}(\epsilon)$, are used to reconstruct the DSD using the method described in Sec. III (i.e., Step 2). Each of the N_f GPR models involved in Step 1 is trained using data from 38 numerical simulations [56] spanning the input space $\mathcal{X} = [10, 65] \times [20, 59] \times [18, 89]$, Fig. 6(c). Specifically, we first run a batch of 28 numerical experiments, of which 20 according to a Latin hypercube sampling and 8 simulations at the edges

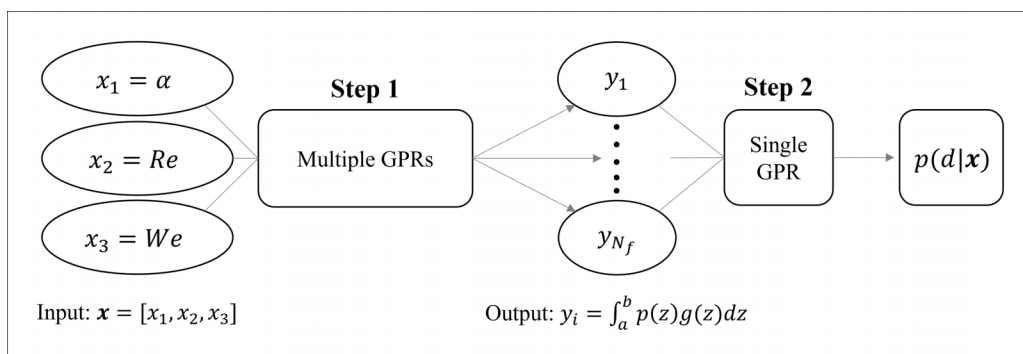


FIG. 5. Schematic of the structure of the surrogate model that maps each parameter, $\mathbf{x} = [x_1, x_2, x_3]$, to the predicted DSD $p(d|\mathbf{x})$. The first step predicts how each feature y_i of the DSD changes with the input by training multiple GPR models, one for each feature. In the second step the predicted features are used as data to reconstruct the DSD with one final GPR model, which uses integral observations.

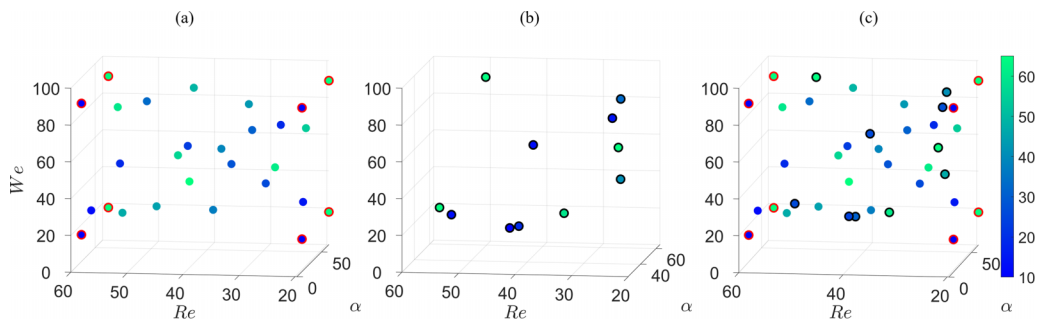


FIG. 6. Design of experiments for the training data. (a) First batch of numerical simulations as points in the input space obtained from a Latin hypercube design and the corners of the cube (red circles). (b) Second batch of 10 simulations identified by minimizing the posterior signal-to-noise ratio. (c) Entire set of simulations performed. The colorbar refers to the value of the spray angle α .

of the three-dimensional input space, see Fig. 6(a). At this point, N_f GPR models are trained to learn each feature y_i based on available data. These models are used to design a second batch of 10 additional numerical experiments, Fig. 6(b). The criterion used for the DOE is the minimization of the signal-to-variance ratio of the predicted $y_i(\mathbf{x})$ within the interrogation region \mathcal{X} . More details on this iterative GPR-based method for the optimal design of experiments can be found in the work by Geng *et al.* [54].

Within Step 1, the i th GPR model is trained using the squared exponential kernel \tilde{k}_i for $i = 1, \dots, N_f$ with automatic relevance determination [57]. The kernel is defined as

$$\tilde{k}_i(\mathbf{x}_m, \mathbf{x}_n) = \tilde{\sigma}_{f,i}^2 \exp\left(-\frac{1}{2}(\mathbf{x}_m - \mathbf{x}_n)^T M_i (\mathbf{x}_m - \mathbf{x}_n)\right) \quad (27)$$

with hyperparameters $\theta_i = \{M_i, \tilde{\sigma}_{f,i}^2\}$, where $M_i = \text{diag}(\mathbf{l}_i)^{-2}$ and $\mathbf{l}_i = [l_i^{(\alpha)}, l_i^{(\text{Re})}, l_i^{(\text{We})}]$. Physically, each element of \mathbf{l}_i plays the role of a characteristic length scale indicating the degree to which a change in the associated design parameter (i.e., α , Re, We) is necessary to induce a significant change in the i th feature. To ease the comparison between length scales, the input space is scaled before training so that all data points are contained in $\mathcal{X}' = [0, 1] \times [0, 1] \times [0, 1]$, also referred to as *min-max* normalization [58].

B. Results

1. Step 1: Performance and sensitivity to design parameters

Table III reports the values of the characteristic length scales associated with the i th feature, \mathbf{l}_i for $i = 1, \dots, 9$, i.e., for the eight cumulative probabilities (p_i for $i = 1, \dots, 8$) plus the first moment (μ_1). We observe that features are more sensitive to changes in We, which, on average, is associated with smaller values of $l_i^{(\text{We})}$, followed by the spray angle α and then Re. In particular, the average diameter decreases as We increases, as visible by comparing Figs. 1(c) and 1(d) with Figs. 1(e) and 1(f), but it is virtually unaffected by Re, as suggested by the large value of $l_9^{(\text{Re})}$. The performance of each of the N_f GPR models involved in Step 1 is evaluated by performing leave-one-out and twofold cross-validation and by computing the associated normalized mean-squared error, MSE_{LOO} and $\text{MSE}_{\text{twofold}}$, respectively. Their values, expressed in percentages of the mean predicted value, are reported in the last two columns of Table III. The average of MSE_{LOO} amongst all the N_f features is 1.25%, showing that the models fit the data well. Notably, the models continue to perform well when tested by twofold cross-validation, i.e., when half of the data points are used for training and half for testing, which result in an average value of $\text{MSE}_{\text{twofold}}$ approximately 1.89%. These observations demonstrate not only that the N_f GPR models provide accurate predictions using only 19 randomly

TABLE III. Characteristic length scales, $l_i^{(\alpha)}$, $l_i^{(\text{Re})}$, and $l_i^{(\text{We})}$, associated with each feature, p_i ($i = 1, \dots, 8$) and μ_1 , along the dimension of the normalized design space \mathcal{X}' . Extremes of the bin associated with the probabilities p_i , b_i , and b_{i+1} . Mean-squared error expressed as percentage of the mean predicted value based on leave-one-out and twofold cross-validations, MSE_{LOO} and $\text{MSE}_{\text{twofold}}$, respectively. Comparing $l_i^{(\alpha)}$, $l_i^{(\text{Re})}$, and $l_i^{(\text{We})}$ we note that We is the most influential parameter and that Re has a negligible effect on the mean drop size. The similar values of $\text{MSE}_{\text{twofold}}$ to MSE_{LOO} shows that performances do not deteriorate substantially when the training data set is halved.

Feature	$l_i^{(\alpha)}$	$l_i^{(\text{Re})}$	$l_i^{(\text{We})}$	b_i	b_{i+1}	MSE_{LOO} (%)	$\text{MSE}_{\text{twofold}}$ (%)
$L_1 = p_1$	2.8	29.7	9.0	0.085	0.146	0.06	0.05
$L_2 = p_2$	5.1	14.2	2.0	0.146	0.213	0.43	0.58
$L_3 = p_3$	9.1	5.7	1.6	0.213	0.287	0.35	0.41
$L_4 = p_4$	4.0	6.1	1.9	0.287	0.368	0.73	0.93
$L_5 = p_5$	6.8	23.0	3.2	0.368	0.457	0.91	0.96
$L_6 = p_6$	2.1	18.0	2.3	0.457	0.555	1.66	7.18
$L_7 = p_7$	1.6	7.9	1.3	0.555	0.663	1.85	2.23
$L_8 = p_8$	2.3	14.1	0.9	0.663	0.794	5.20	4.46
$L_9 = \mu_1$	3.3	71.6	2.6			0.05	0.15

sampled data points but also that they do not suffer overfitting when all the 38 simulations are used for training.

2. Step 2: Exploring the design space

By combining the two steps in Fig. 5, we explore the design space and produce DSDs at different values of We, Re, and α , as reported in Figs. 7(a), 8(a), and 9(a), respectively. Each DSD has its maximum near the minimum diameter resolved in the numerical simulations, followed by a decrease of the probability with increasing drop diameter. This leads to a plateau-like region of intermediate-sized drops where the probability tends to decrease less steeply, followed by the tail. As expected from considerations of the instability of the liquid sheet leading to ligament formation [11, 12], increasing We leads to a faster decay of the DSD tail and a corresponding increase in the probability at small and intermediate diameters. The reduction in We also shifts the peak of the DSD towards larger diameters, as shown in the inset of Fig. 7(a), in agreement with experimental evidence [29]. We remark that the SM can qualitatively capture such a shift in the peak despite the

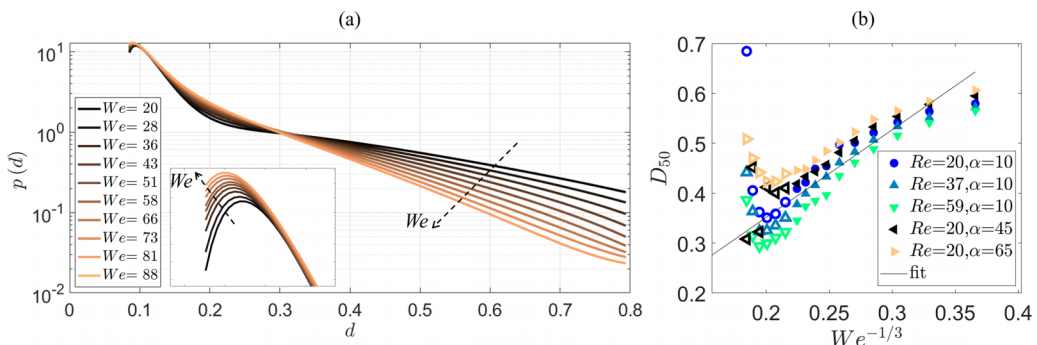


FIG. 7. (a) Drop size distributions predicted by the SM at different values of the Weber number (inset: close-up of the peaks). (b) Volume mean diameter D_{50} as a function of $We^{-1/3}$ for different spray angles and Reynolds numbers. Open symbols are in extrapolation region.

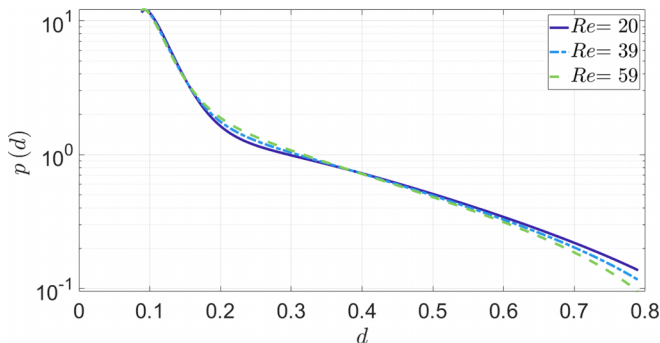


FIG. 8. Drop size distributions predicted by the SM at different values of the Reynolds number.

fact that it is located in a low-confidence region (i.e., small drops) and that the same binning scheme, reported in Table III, has been used for all the simulations.

The scaling $D_{50} \sim We^{-1/3}$ of the volume mean diameter D_{50} predicted by Kooij *et al.* [29] is also tested for different values of the Reynolds number and the spray angle, see Fig. 7(b). The values of D_{50} are computed from synthetic populations of drops generated via a pseudorandom number generator from the DSDs predicted by the SM. Towards the right side of the plot the trend starts to be nonlinear which could be signaling a change on the relative importance of different breakup mechanisms at small values of the Weber numbers, which are several order of magnitude smaller than those for which the scaling was observed. On the left side of the figure are reported the predicted D_{50} in the extrapolation region, i.e., for $We > 90$ (empty symbols). Our results show that the SM manages to predict the correct scaling up to $We^{-1/3} \approx 0.20$, thus corresponding to a 40% increase with respect to the highest We in the training data. Qualitatively, both D_{50} and the characteristics of the DSD appear to be affected in a similar way by the Reynolds number, even if it has a quantitative smaller effect (Fig. 8).

Finally, we observe that the main effect of increasing the spray angle is to increase the probability of drops of intermediate size up until the tail of the distribution, as shown in Fig. 9(a). To better interpret this result, we use the DSDs to produce equally sized synthetic populations of $N_d = O(10^4)$ drops, $\mathcal{S} = \{d_i\}_{i=1}^{N_d}$, for different values of α , Re , and We . Then each diameter d_i is converted into the equivalent volumes $v_i = \pi d_i^3/6$, and the populations of volumes are finally used to compute the

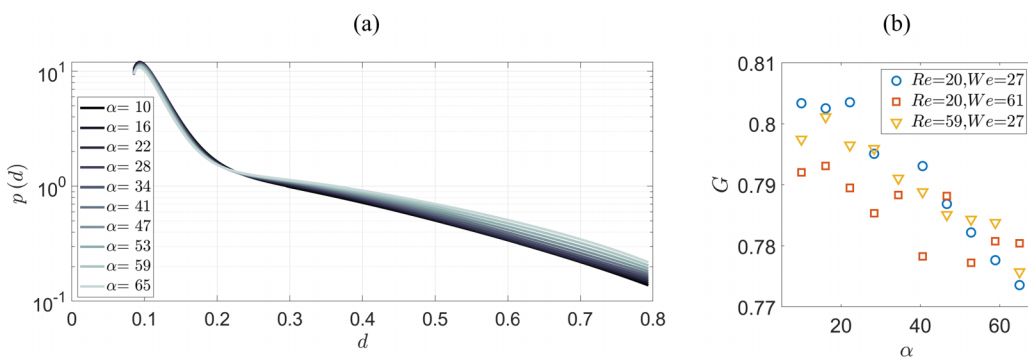


FIG. 9. (a) Drop size distributions predicted by the SM at different values of the spray angle α . (b) Gini index of the volume distribution as a function of α .

Gini index,

$$G = \frac{\sum_{i=1}^{N_d} \sum_{j=1}^{N_d} |v_i - v_j|}{2N_d^2 \bar{v}}, \quad (28)$$

where \bar{v} is the arithmetic mean of the population of volumes. The Gini index, originally developed to measure the income inequality within a given social group [59], is used here to measure, within a given spray, the inequality in volume (i.e., mass) distribution of drops. A Gini index of one corresponds to a spray in which all the volume is in one single drop, while a Gini index of zero corresponds to the case where the volume of the spray is distributed in equally sized drops. Our results, reported in Fig. 9(b), show that a larger spray angles correspond to smaller G , thus to more equally distributed populations in terms of the volume in each drop.² This result can be explained by noting that the drops formed at the rims of the spray are larger than those generated by the flapping instability of the liquid sheet [11,29], see Figs. 1(c) and 1(d). The relative importance of these larger drops increases in narrow sprays, leading to a more unequal distribution of the volume and, thus, larger G .

V. CONCLUSIONS

We propose a quantitatively accurate data-driven method to predict the DSD of a flat fan spray as a function of the spray angle and working conditions, which are parameterized by the Reynolds and Weber numbers of the jet. The method is based on GPR models; therefore, the prediction has confidence intervals and takes into account the uncertainty of the data. Key to our method is the observation of integrals (i.e., cumulative probabilities) and the first moment (i.e., expected value) to reconstruct the unknown probability density function. These features of the DSD determine the percentage of atomized fluid contained in droplets of a certain size and the bias of the estimator. First, we show that integral observations for the inference of the continuous DSD provides physically consistent and accurate predictions, which overcome the limitations of the commonly used point estimates of the DSD. Second, we show that the training and predictions of the proposed model are robust with respect to the selection of the numerical simulations used for training and test and the model's hyperparameters. Third, we propose kernels, which are physically motivated. The DSD is estimated without assuming its shape *a priori*, such as that of Gamma or log-normal distributions, as typically performed in the literature. We embed the physical fact that liquid atomization gives rise to sharply peaked, heavy-tailed distribution into the choice of the covariance function. Our analysis demonstrates that the SM is versatile for the prediction of more intricate (e.g., bimodal) distributions, while retaining the sharply peaked, heavy-tailed characteristics.

The proposed SM can be adapted to characterize other complex processes, the outcome of which is a probability density function that changes with the control parameters. For example, these include other multiphase problems, such as the modeling of the bubble size distribution in chemical reactors. The application of this approach to experimental data and larger parameter spaces is the subject of current and future work. The data from the 38 numerical simulations used in this study for the training of the SM are available [56].

ACKNOWLEDGMENT

We acknowledge funding from the Engineering and Physical Sciences Research Council, UK, through the Programme Grant PREMIERE (EP/T000414/1). L.M. also acknowledges financial

²The same analysis could be performed in term of linear size, i.e., diameters d_i , in which case we could compute G directly from the continuous DSD. We choose to use the volume as directly related to the mass distribution, thus more practically relevant.

support from the ERC Starting Grant PhyCo 949388 and the UKRI AI for Net Zero Grant No. EP/Y005619/1.

- [1] S. Li, C. Chen, Y. Wang, F. Kang, and W. Li, Study on the atomization characteristics of flat fan nozzles for pesticide application at low pressures, *Agriculture* **11**, 309 (2021).
- [2] A. H. Lefebvre, The role of fuel preparation in low-emission combustion, *J. Eng. Gas Turbines Power* **117**, 617 (1995).
- [3] A. H. Lefebvre and V. G. McDonell, *Atomization and Sprays* (CRC Press, Boca Raton, FL, 2017)
- [4] M. C. B. Ellis, C. R. Tuck, and P. C. H. Miller, The effect of some adjuvants on sprays produced by agricultural flat fan nozzles, *Crop Protection* **16**, 41 (1997).
- [5] P. A. Hobson, P. C. H. Miller, P. J. Walklate, C. R. Tuck, and N. M. Western, Spray drift from hydraulic spray nozzles: the use of a computer simulation model to examine factors influencing drift, *J. Agricult. Eng. Res.* **54**, 293 (1993).
- [6] E. Babinsky and P. E. Sojka, Modeling drop size distributions, *Prog. Energy Combust. Sci.* **28**, 303 (2002).
- [7] B. Mohan, W. Yang, and S. C. Kiang, Fuel injection strategies for performance improvement and emissions reduction in compression ignition engines—a review, *Renew. Sust. Energy Rev.* **28**, 664 (2013).
- [8] R. Schmehl, G. Maier, and S. Wittig, CFD analysis of fuel atomization, secondary droplet breakup and spray dispersion in the premix duct of a LPP combustor, in *Proceedings of the 8th International Conference on Liquid Atomization and Spray Systems (ICLASS'00)* (ILASS, Pasadena, California, USA, 2000).
- [9] L. Andreassi, S. Ubertini, and L. Allocca, Experimental and numerical analysis of high pressure diesel spray-wall interaction, *Int. J. Multiphase Flow* **33**, 742 (2007).
- [10] L. Wang, N. Zhang, J. W. Slocombe, G. E. Thierstein, and D. K. Kuhlman, Experimental analysis of spray distribution pattern uniformity for agricultural nozzles, *Appl. Eng. Agric.* **11**, 51 (1995).
- [11] E. Villermaux and C. Clanet, Life of a flapping liquid sheet, *J. Fluid Mech.* **462**, 341 (2002).
- [12] E. Villermaux, Fragmentation, *Annu. Rev. Fluid Mech.* **39**, 419 (2007).
- [13] A. Khopkar, M. D. Cloeter, and Q. Yuan, Industrial sprays: Experimental characterization and numerical modeling, in *Applied Computational Fluid Dynamics* (IntechOpen, London, UK, 2012).
- [14] C. R. Constante-Amores, L. Kahouadji, A. Batchvarov, S. Shin, J. Chergui, D. Juric, and O. K. Matar, Direct numerical simulations of transient turbulent jets: Vortex-interface interactions, *J. Fluid Mech.* **922**, A6 (2021).
- [15] Y. Ling, D. Fuster, S. Zaleski, and G. Tryggvason, Spray formation in a quasiplanar gas-liquid mixing layer at moderate density ratios: A numerical closeup, *Phys. Rev. Fluids* **2**, 014005 (2017).
- [16] S. Shin, J. Chergui, D. Juric, L. Kahouadji, O. K. Matar, and R. V. Craster, A hybrid interface tracking–level set technique for multiphase flow with soluble surfactant, *J. Comput. Phys.* **359**, 409 (2018).
- [17] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, Design and analysis of computer experiments, *Stat. Sci.* **4**, 409 (1989).
- [18] K. Chaloner and I. Verdinelli, Bayesian experimental design: A review, *Stat. Sci.* **10**, 273 (1995).
- [19] T. P. Sapsis, Output-weighted optimal sampling for Bayesian regression and rare event statistics using few samples, *Proc. R. Soc. A* **476**, 20190834 (2020).
- [20] S. Conti and A. O’Hagan, Bayesian emulation of complex multi-output and dynamic computer models, *J. Stat. Plan. Inference* **140**, 640 (2010).
- [21] N. Bempedelis and L. Magri, Bayesian optimization of the layout of wind farms with a high-fidelity surrogate model, *arXiv:2302.01071*.
- [22] K. Campbell, M. D. McKay, and B. J. Williams, Sensitivity analysis when model outputs are functions, *Reliab. Eng. Syst. Safe.* **91**, 1468 (2006).
- [23] M. J. Bayarri, D. Walsh, J. O. Berger, J. Cafeo, G. Garcia-Donato, F. Liu, J. Palomo, R. J. Parthasarathy, R. Paulo, and J. Sacks, Computer model validation with functional output, *Ann. Stat.* **35**, 1874 (2007).

- [24] K. Longi, C. Rajani, T. Sillanpaa, J. Makinen, T. Rauhala, A. Salmi, E. Haeggström, and A. Klami, Sensor placement for spatial Gaussian processes with integral observations, in *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI'20)*, Proceedings of Machine Learning Research, Vol. 124, edited by J. Peters and D. Sontag (PMLR, 2020), pp. 1009–1018.
- [25] H. C. Law, D. Sejdinovic, E. Cameron, T. Lucas, S. Flaxman, K. Battle, and K. Fukumizu, Variational learning on aggregate outputs with Gaussian processes, *Advances in Neural Information Processing Systems*, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Curran Associates, Inc., 2018), Vol. 31.
- [26] E. Solak, R. Murray-Smith, W. E. Leithead, D. Leith, and C. Rasmussen, Derivative observations in Gaussian process models of dynamic systems, *Advances in Neural Information Processing Systems*, edited by S. Becker, S. Thrun, and K. Obermayer (MIT Press, 2002), Vol. 15.
- [27] Q. Cao, G. Zhang, E. A. Brandes, and T. J. Schuur, Polarimetric radar rain estimation through retrieval of drop size distribution using a Bayesian approach, *J. Appl. Meteorol. Climatol.* **49**, 973 (2010).
- [28] S. Tengeling and N. Armand, Performance of using cascade forward back propagation neural networks for estimating rain parameters with rain drop size distribution, *Atmosphere* **5**, 454 (2014).
- [29] S. Kooij, R. Sijts, M. M. Denn, E. Villermaux, and D. Bonn, What determines the drop size in sprays? *Phys. Rev. X* **8**, 031019 (2018).
- [30] C. K. Williams and C. E. Rasmussen, *Gaussian Processes for Machine Learning*, Vol. 2 (MIT Press, Cambridge, MA, 2006).
- [31] GPy, GPy: A Gaussian process framework in python [<http://github.com/SheffieldML/GPy>].
- [32] J. Gardner, G. Pleiss, K. Q. Weinberger, D. Bindel, and A. G. Wilson, GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration, *Advances in Neural Information Processing Systems*, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Curran Associates, Inc., 2018), Vol. 31.
- [33] M. A. Alvarez, L. Rosasco, N. D. Lawrence *et al.*, Kernels for vector-valued functions: A review, *Found. Trends Mach. Learn.* **4**, 195 (2012).
- [34] A. O'Hagan, Bayes-hermite quadrature, *J. Stat. Plan. Inference* **29**, 245 (1991).
- [35] C. E. Rasmussen and Z. Ghahramani, Bayesian Monte Carlo, *Advances in Neural Information Processing Systems*, edited by S. Becker, S. Thrun, and K. Obermayer (MIT Press, 2003), Vol. 15.
- [36] T. Gunter, M. A. Osborne, R. Garnett, P. Hennig, and S. J. Roberts, Sampling for inference in probabilistic models with fast Bayesian quadrature, *Advances in Neural Information Processing Systems*, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (Curran Associates, Inc., 2014), Vol. 27.
- [37] J. A. Rice, *Mathematical Statistics and Data Analysis* (Cengage Learning, Independence, KY, 2006).
- [38] T. Abadie, J. Aubin, and D. Legendre, On the combined effects of surface tension force calculation and interface advection on spurious currents within volume of fluid and level set frameworks, *J. Comput. Phys.* **297**, 611 (2015).
- [39] S. Popinet, Numerical models of surface tension, *Annu. Rev. Fluid Mech.* **50**, 49 (2018).
- [40] S. Popinet, A quadtree-adaptive multigrid solver for the Serre–Green–Naghdi equations, *J. Comput. Phys.* **302**, 336 (2015).
- [41] S. Popinet, An accurate adaptive solver for surface-tension-driven interfacial flows, *J. Comput. Phys.* **228**, 5838 (2009).
- [42] G. Tryggvason, R. Scardovelli, and S. Zaleski, *Direct Numerical Simulations of Gas–Liquid Multiphase Flows* (Cambridge University Press, Cambridge, UK, 2011).
- [43] W. Aniszewski, T. Ménard, and M. Marek, Volume of fluid (VOF) type advection methods in two-phase flow: A comparative study, *Comput. Fluids* **97**, 52 (2014).
- [44] S. Afkhami and M. Bussmann, Height functions for applying contact angles to 2D VOF simulations, *Int. J. Numer. Methods Fluids* **57**, 453 (2008).
- [45] E. G. Puckett and J. S. Saltzman, A 3D adaptive mesh refinement algorithm for multimaterial gas dynamics, *Physica D* **60**, 84 (1992).
- [46] J. L. Nowak, M. Mohammadi, and S. P. Malinowski, Applicability of the VisiSize D30 shadowgraph system for cloud microphysical measurements, *Atmos. Meas. Tech.* **14**, 2615 (2021).

- [47] *VisiSize Software Operation Manual-VisiSize 6* (Oxford Lasers Inc., 2013).
- [48] Y. Jin, F. R. Beck, B. R. Lowery, D. J. Miller, F. Cheung, S. M. Bajorek, K. Tien, and C. L. Hoxie, Experimental study of droplet sizes across a spacer grid location under various reflow conditions, *Exp. Therm. Fluid Sci.* **94**, 246 (2018).
- [49] P. Marmottant and E. Villermaux, Fragmentation of stretched liquid ligaments, *Phys. Fluids* **16**, 2732 (2004).
- [50] P. Marmottant and E. Villermaux, On spray formation, *J. Fluid Mech.* **498**, 73 (2004).
- [51] R. D. Cohen, Shattering of a liquid drop due to impact, *Proc. R. Soc. Lond. A* **435**, 483 (1991).
- [52] M. N. Gibbs, Bayesian Gaussian processes for regression and classification, Ph.D. thesis, Citeseer (1998).
- [53] A. Conn, N. Gould, and P. Toint, A globally convergent Lagrangian barrier algorithm for optimization with general inequality constraints and simple bounds, *Math. Comput.* **66**, 261 (1997).
- [54] Z. Geng, F. Yang, X. Chen, and N. Wu, Gaussian process based modeling and experimental design for sensor calibration in drifting environments, *Sens. Actuat. B: Chem.* **216**, 321 (2015).
- [55] J. S. Simonoff, *Smoothing Methods in Statistics*, Springer Series in Statistics (Springer-Verlag, New York, 1996).
- [56] T. Traverso, T. Abadie, O. K. Matar, and L. Magri, Drop count and size of 38 numerical simulation of a flat fan spray [Data set], *Zenodo* (2023).
- [57] R. M. Neal, Bayesian learning for neural networks, Ph.D. thesis, University of Toronto, 1994.
- [58] W. Li and L. Zhenyu, A method of SVM with normalization in intrusion detection, *Proc. Environ. Sci.* **11**, 256 (2011).
- [59] J. A. Weymark, Generalized gini inequality indices, *Math. Soc. Sci.* **1**, 409 (1981).