AIAA Journal
Vol. 57, No. 12, December 2019

Kernel-Smoothed Proper Orthogonal Decomposition–Based Emulation for Spatiotemporally Evolving Flow Dynamics Prediction

Yu-Hung Chang,* Liwei Zhang,† Xingjian Wang,‡ Shiang-Ting Yeh,§ Simon Mak,¶ Chih-Li Sung,¶
C. F. Jeff Wu,** and Vigor Yang††
Georgia Institute of Technology, Atlanta, Georgia 30332

DOI: 10.2514/1.J057803

This interdisciplinary study, which combines machine learning, statistical methodologies, high-fidelity simulations, projection-based model reduction, and flow physics, demonstrates a new process for building an efficient surrogate model to predict spatiotemporally evolving flow dynamics for design survey. In our previous work, a common proper-orthogonal-decomposition (CPOD) technique was developed to establish a physics-based surrogate (emulation) model for prediction of useful flow physics and design exploration over a wide parameter space. The emulation technique is substantially improved upon here using a kernel-smoothed POD (KSPOD) technique, which leverages kriging-based weighted functions from the design matrix. The resultant emulation model is then trained using a large-scale dataset obtained through high-fidelity simulations. As an example, the flow evolution in a swirl injector is considered for a wide range of design parameters and operating conditions. The KSPOD-based emulation model performs well and can faithfully capture the spatiotemporal flow dynamics. The model enables effective design surveys using high-fidelity simulation data, achieving a turnaround time for evaluating new design points that is 42,000 times faster than the original simulation.

Nomenclature

A_{in}	=	total inlet area
A_n	=	cross-sectional area
$\boldsymbol{c}$	=	design setting (parameter set)
D	=	Sobol response variance over design range
f	=	flow property
h	=	liquid film thickness at injector exit
$\boldsymbol{K}$	=	geometric constant
L	=	injector length
p	=	design parameters
R	=	Gaussian correlation function
R_n	=	injector radius
T	=	temperature
t	=	time
w	=	kriging weighing number
$\boldsymbol{x}$	=	spatial coordinate
Z	=	zero-mean Gaussian process
α	=	liquid film spreading angle at injector exit
β_j	=	time-varying coefficients
δ	=	inlet slot width
ΔL	=	distance between injector inlet and headend
μ	=	mean
ϕ_j	=	spatial basis functions
ρ	=	density
θ	=	tangential inlet angle

Subscripts

n	=	number of simulations
u	=	subset of design parameters $u \subseteq \{1, \ldots, p\}$

I. Introduction

THE purpose of this work is to develop an efficient surrogate model (emulator) for prediction of spatiotemporally evolving flow dynamics. The study involves computational fluid dynamics (CFD), projection-based reduced modeling, statistics, and machine learning. As a demonstration case, the flow evolution in a swirl injector is presented. In recent years, machine learning algorithms have been developed and applied to improve the model accuracy and efficiency for predicting turbulent flow dynamics. For example, Ling and colleagues identified regions of high Reynolds-averaged Navier–Stokes (RANS) uncertainty [1] and subsequently improved the prediction of Reynolds stress anisotropy for this type of flows [2]. Duraisamy et al. developed a paradigm for data-driven modeling of turbulent flows using a machine-learning technique [3,4]. These works initiated efforts to incorporate data science into the study of fluid dynamics.

For design assessment, physical experiments can be extremely expensive and time-consuming, especially for complex systems operating over a wide range of conditions. Moreover, it is hard to gain insight into underlying physicochemical mechanisms through measurements using currently available experimental techniques. To better capture flow characteristics and identify design attributes, one solution is high-fidelity modeling and simulations such as large-eddy simulation (LES). The LES framework employed in the present work is capable of dealing with fluid flow and combustion dynamics over the entire range of thermodynamic states [5–9]. These simulations, however, are computationally expensive and impractical for use as a primary tool to survey the entire design space; an axisymmetric simulation of flow evolution in a simplex swirl injector using LES, for instance, may take about 100,000 CPU hours on the hexa-core AMD Opteron Processor 8431. The traditional trial-and-error–based design practice is no longer practical.

To enable the use of high-fidelity simulations for design evaluation, an effective emulation (surrogate) model must be incorporated into the design process [10–12]. The procedure takes a few steps. First, design of experiments (DoE), which achieves more

Received 8 August 2018; revision received 13 June 2019; accepted for publication 23 July 2019; published online 29 August 2019.

*Co-First Author, Graduate Student, School of Aerospace Engineering.

†Co-First Author, Research Engineer, School of Aerospace Engineering.

‡Research Engineer, School of Aerospace Engineering.

§Graduate Student, School of Aerospace Engineering.

¶Graduate Student, School of Industrial and Systems Engineering.

**Professor and Coca-Cola Chair in Engineering Statistics, School of Industrial and Systems Engineering.

††William R. T. Oakes Professor and Chair, School of Aerospace Engineering; vigor.yang@aerospace.gatech.edu (Corresponding Author).

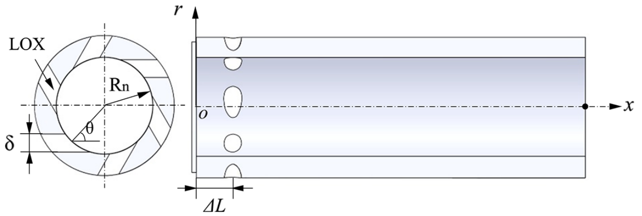

Fig. 1 Schematic of swirl injector.

realistic computation timelines for building a database, is implemented. Here we consider a swirl injector as a demonstration example [13], as shown schematically in Fig. 1. The detailed flow characteristics have been previously explored using LES techniques [14,15]. DoE can be formulated based on several key geometric parameters and their respective ranges of consideration. The present study focuses on the effects of these geometric parameters (i.e., location and width of the tangential entry and injection angle) on the injector performance, as measured by the thickness and spreading angle of the liquid film at the exit of the injector [10,12–15]. The total sample size is determined using a $10d$ rule-of-thumb described by Loeppky et al. [16], which recommends 10 simulations per design parameter where d means the total number of design parameters. This approach substantially reduces the number of total sample points required to survey the design space.

The second step is the creation of a database with sufficient information to allow for a survey of the design space. This can be achieved by performing LES-based high-fidelity simulations at the selected design points from DoE. For spatiotemporally evolving flows, however, the resulting database is too large to be handled effectively. Identification of dominant flow structures and reduction of the "big data" becomes essential for building the emulation model [10]. To this end, a projection-based approach [17], proper orthogonal decomposition (POD) technique, is adopted to extract these coherent structures into a reduced set of orthogonal basis functions.

The final step requires the combination of POD and kriging methodologies. Kriging, regarded as a parametric, data-fitted surrogate model, is a powerful machine-learning tool for interpolation and prediction [18]. The concept of kriging is to model unobserved responses using a Gaussian process (GP) governed by a preset covariance function. The response surface of the training model can be evaluated via data-tuned weights to radial basis functions centered at observed points. In the present study, each simulation contains 380,928 numerical grid points in the spatial domain. Kriging is required for each point if information for each point is to be modeled. To reduce the data size, POD [19] has been incorporated into the prediction model (emulator).

In our previous work [11,12], common-grid POD (CPOD) was implemented to build an emulation model. The method successfully predicts mean flow structures in swirl injectors with a broad range of geometric dimensions. To improve the prediction accuracy of flow evolution in the entire spatial domain and associated flow dynamics, Kernel-smoothed POD (KSPOD) is proposed here. Two main assumptions must be addressed. First, the physics extracted by POD modes in different cases are similar under the same rank, as determined by the POD energy. Second, the dominant modes capturing similar physics are transferred with similar phase and these physics are retained through the kriging-weighted averaging based on the new design. KSPOD achieves three goals: 1) capturing turbulent flow dynamics; 2) predicting results as verified through quantitative comparisons with simulation results; 3) yielding predictions with short turnaround times. The KSPOD-based emulator requires only about 0.02 CPU hour to predict the flowfield at a new design point for the problem presented in the present study. The overall computation time, including data loading and training, is about 25 CPU hours to predict the flow evolution at a new design point for a time duration of 10 ms with 1000 snapshots. For comparison, the corresponding LES calculation takes about 100,000 CPU hours for a spatial domain of 380,928 numerical grid points.

The work described in this paper produces a high-fidelity surrogate modeling technique for efficient prediction of complex flowfields over a broad range of operating conditions and geometric parameters. The paper is structured as follows. Section II introduces the swirl injector configuration, sample points designated by the DoE, the numerical framework, the emulation model, and the overall design process. Section III discusses the simulation and emulation results. The emulator is systematically assessed based on performance metrics, relative error, and spectral information on simulated and predicted flowfields. Section IV concludes with directions for future work.

II. Methodology

A. Swirl Injector Configuration

Figure 1 shows a schematic of the swirl injector of concern. Liquid oxygen (LOX) is tangentially introduced into the injector and develops a swirling film that is attached to the wall due to centrifugal force [14,20]. Conservation of angular momentum results in a hollow gaseous core in the center region. The liquid film exits the injector as a thin conical sheet and subsequently undergoes atomization into droplets. The flow dynamics in this type of device under supercritical conditions have been extensively studied using LES techniques [15,16].

The selection of design variables is dependent on system requirements. In our previous studies [11–13], the design consisted of five parameters: injector length, L; injector radius, R_n; tangential inlet width, δ; tangential injection angle, θ; and distance between the center of inlet and headend, ΔL. The sample points were chosen based on the maximum projection (MaxPro) [21] technique across a broad design space. The injector length covers a range of 22–93 mm. Sensitivity analysis is used to identify the most significant parameters dictating injector performance. Our previous work [11] shows that the injection width (δ) is the most important parameter in the determination of the spreading angle of the liquid film. The tangential inlet angle (θ) and the injection width (δ) significantly affect the liquid film thickness, whereas the injector length L and radius R_n play minor roles. Thus, the present work focuses on the injection width δ, and angle θ, and the distance between the inlet and headend ΔL.

Table 1 tabulates the baseline geometry and operating conditions, including the LOX inlet temperature T_{in}, ambient temperature T_∞, ambient pressure p_∞, and mass flow rate $\dot{m}$. Another parameter is the geometric constant $\boldsymbol{K}$, a nondimensional parameter that can be used to evaluate the flow characteristics of swirl injectors, namely, the liquid film thickness and spreading angle. The former controls film atomization and the latter mixing efficiency [14]. The geometric constant takes the following definition:

$$\boldsymbol{K} = A_n R_{\mathrm{in}} / A_{\mathrm{in}} R_n \tag{1}$$

where A_n denotes the cross-sectional area of the injection exit and A_{in} the total inlet area. The geometric constant is an indicator of the swirl strength. When the value is high, a large angular momentum is present in the liquid film, leading to a wide spreading angle. The liquid film thickness and spreading angle must be defined carefully. At subcritical pressure, a distinct interface exists between the liquid and gas phases. At supercritical pressure, however, density decreases smoothly as the fluid transits from liquid to its supercritical state. A pseudo-boiling point, where the maximum density gradient occurs, is used to characterize the interface. The thickness and spreading angle of the liquid film at the injector exit are thus measured based on the location of the maximum density gradient in the present study [15].

Table 2 shows the design space and the range for each design variable. The distance between the inlet and headend, ΔL, is decided by a rule of thumb to be 1.5–2.0 times the injector width [14]. This is

Table 1 Baseline geometry and operating conditions

R, mm	R_{in}, mm	L, mm	$\dot{m}$, kg/s	T_{in}, K	T_∞, K	p_∞, MPa
4.50	0.85	25	0.17	120	300	10

Table 2 Design space

Design variable	θ, deg	δ, mm	ΔL, mm
Design range	35.0–62.2	0.27–1.53	0.85–3.40

an optimal location determined from a tradeoff study to avoid 1) excessive viscous losses when the injection slit is too close to the headend and 2) low-frequency oscillations due to the presence of a large recirculation zone if the inlet is too far from the headend. The design space of injection width δ and angle θ is decided by the desired range of spreading angle (50°–62°) and film thickness (0.66–1.50 mm). With these numbers and the geometric constant, the ranges of δ and θ can be estimated.

B. High-Fidelity Simulation

The theoretical formulation for high-fidelity simulations, which treats supercritical fluid flows and combustion over the entire range of fluid thermodynamic states of concern, is described in detail in [7,8]. Turbulence closure is achieved by means of LES techniques. The effects of subgrid-scale motion are represented by an improved Smagorinsky eddy-viscosity model. Thermodynamic properties are evaluated according to fundamental thermodynamics theories and a modified Soave–Redlich–Kwong (SRK) equation of state (EOS). The Takahashi method calibrated for high-pressure conditions is employed to obtain the mass diffusivity. Transport properties are evaluated using extended corresponding-state principles.

The numerical framework is based on a preconditioning scheme with a unified treatment of general-fluid thermodynamics [22,23]. It applies a density-based, finite-volume methodology, along with a dual-time-step integration technique [24]. A second-order backward difference is used to accomplish temporal discretization, and a four-step Runge–Kutta scheme is applied to integrate the inner-loop pseudo-time term. A fourth-order central difference scheme in generalized coordinates is used to obtain spatial discretization. Fourth-order matrix dissipation is taken to assure numerical stability and minimum contamination of the solution. Lastly, a multiblock domain decomposition technique associated with the message passing interface technique for parallel computing is applied to optimize computation speed.

Owing to the demanding computational requirements of three-dimensional simulations, only a cylindrical sector of the swirl injector is considered in the simulations, with periodic boundary conditions specified in the azimuthal direction. At the LOX inlet, an acoustically nonreflecting boundary condition is implemented [25]. At the downstream boundary of the computational domain, sponge layers are employed in both the axial and radial directions to prevent the reflection of numerical disturbances [26]. All solid walls are assumed to be adiabatic, and no-slip conditions are enforced.

The theoretical/numerical framework has previously been validated against a variety of supercritical fluid flow problems [15,27–30]. To minimize numerical uncertainty, a grid-convergence study was performed for the baseline design. Three different grid levels were considered, with a refinement ratio of two in each direction. In consideration of computational accuracy and efficiency, the intermediate grid with 380,928 finite-volume cells (184,320 in the injector and 196,608 in the downstream region) is selected for all the computations in the present study. The time step is fixed at 2.0×10^{-7} s for temporal accuracy. The local Courant-Friedrichs-Lewy number varies in the range of 0.1–0.7, depending on local flow velocities and grid sizes. Each LES calculation takes about 100,000 CPU hours on the hexa-core AMD Opteron Processor 8431 to obtain statistically significant data. The simulation data are recorded over a physical time span of 10 ms, after the flowfield has reached its fully developed state. A total of 1000 snapshots with a time interval of 10 μs are collected for each case. According to the Nyquist criterion, a frequency resolution of 50 kHz is achieved for the collected data set, covering all important characteristic frequencies of flow dynamics in the present swirl injector.

C. Design of Experiments

The DoE methodology is a statistical approach for careful selection of input variables for a given design space. DoE facilitates the design process and reduces the number of total sample points required to efficiently explore the design space. With prior knowledge of the major contributing geometric parameters, the sample size is determined based on the $10d$ rule-of-thumb [17], where d denotes the total number of design parameters. The present study considers three design variables (θ, δ, and ΔL). Although MaxPro provides good space-filling properties on both the design space and its projections [21], MaxPro does not provide a sequential design capability, giving optimal space-filling performance in batches. To that end, sliced Latin hypercube design (SLHD) [31–33] is selected. In SLHD, the space-filling performance of the design points in each slice is optimal. The overall design matrix contains five slices, and each slice includes six design points. Figure 2 shows the two-dimensional projections of the design points categorized by different slices. Each case takes about 100,000 CPU hours for high-fidelity simulation to obtain statistically significant data. A total of 1000 snapshots spanning 10 ms are acquired after the flowfield reaches its stationary state (~12 ms). The snapshots are subsampled every 20 temporal iterations, each with a time step of 0.5 μs. A temporal resolution of 50 kHz is achieved, according to the Nyquist criterion.

D. Kernel-Smoothed POD

This section introduces kernel-smoothed proper orthogonal decomposition (KSPOD), which combines statistical modeling with a projection-based method to obtain a reduced-basis surrogate model. The proposed KSPOD method can be viewed as a generalization of the POD used for flow emulation. POD decomposes the flowfield into an expansion consisting of spatial eigenfunctions, known as POD modes, and corresponding time-varying coefficients. Such a decomposition can be written in the following form:

$$f(\boldsymbol{u}, t) \approx \sum_{k=1}^{K} \beta^k(t)\phi^k(\boldsymbol{u}) \tag{2}$$

where $f(\boldsymbol{u}, t)$ is the simulated flowfield at spatial location $\boldsymbol{u}$ and time t, and $\beta^k(t)$ and $\phi^k(\boldsymbol{u})$ represent the time-varying coefficient and basis function for the kth mode, respectively. As indicated in Eq. (2), the

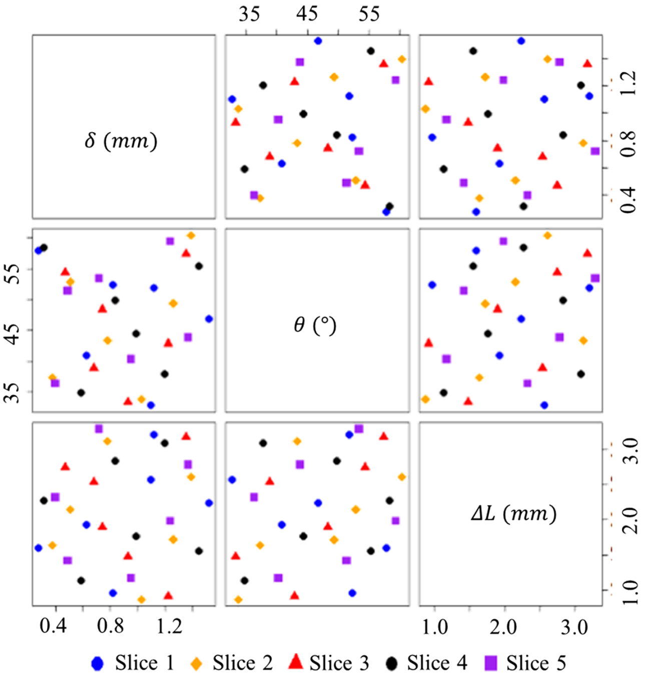

Fig. 2 Two-dimensional projection of design points obtained by sliced Latin hypercube design methodology in the design space.

expansion is truncated with the first K terms, where K is chosen such that the reconstructed flowfield retains a desired degree of accuracy. In practice, the time-varying coefficients and basis functions are obtained through an eigen-decomposition of the inner product of a flowfield variable [19]. Equation (2) can be viewed as the optimal decomposition of $f(\boldsymbol{u}, t)$ using a basis expansion of K terms.

From a physics perspective, POD provides valuable insight into the important physics present in the flowfield. The basis function, or mode shape, $\phi^k(\boldsymbol{u})$ can be interpreted as spatial distribution of the fluctuation field of a given flow variable (such as pressure, density, temperature, and velocity components). It represents the dominant coherent structure, such as acoustic waves in the system [34]. A spectral analysis of the POD coefficients can be performed to identify flow periodicity and characteristic frequencies for hydrodynamic and acoustic instabilities. The index for the basis-function expansion in Eq. (2) is determined by the rank of the energy content in the eigen-decomposition calculation and suggests which flow structure is more prevalent. The first few terms in the expansion represent more energy-containing structures, and the remaining terms represent increasingly weaker flow features.

The goal here is to employ the flow features extracted using POD within a statistical framework, allowing the training of an emulator for flow prediction. To this end, we employ a machine-learning technique called Gaussian process regression, also known as kriging, to predict both POD modes and time-varying coefficients at a new design setting. We begin with a brief introduction into the mathematical formulation behind kriging, and then describe how such a model is incorporated into the KSPOD framework.

Kriging is a powerful learning technique that leverages a Gaussian process regression model to learn the structure of an unknown function by sampling this function at specific points. The mathematical framework for kriging is established as described in [35,36]. We assume that the unknown function of interest (i.e., POD modes and time-varying coefficients at a new design point in the present study), $Y(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^d$, is a realization from the stochastic process

$$Y(\boldsymbol{x}) = \mu + Z(\boldsymbol{x}) \tag{3}$$

where $\boldsymbol{x}$ is a d-dimensional vector, μ is the mean of the process, and $Z(\boldsymbol{x})$ is a zero-mean Gaussian process with $\mathrm{Var}\{Z(\boldsymbol{x})\} = \sigma^2$ and correlation function

$$\mathrm{Corr}[Z(\boldsymbol{x}_i), Z(\boldsymbol{x}_j)] = R(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{4}$$

where $R(\boldsymbol{x}_i, \boldsymbol{x}_j)$ denotes the correlation between the random variables $Z(\boldsymbol{x}_i)$ and $Z(\boldsymbol{x}_j)$. Following common practice [37], we employ the squared-exponential correlation function:

$$R(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left[-\sum_{k=1}^{d} \theta_k (\boldsymbol{x}_{ik} - \boldsymbol{x}_{jk})^2\right] \tag{5}$$

where $\boldsymbol{x}_{ik}$ is the kth element of $\boldsymbol{x}_i$ and θ the correlation parameter tuned by maximum-likelihood estimation.

A key advantage of kriging is that a closed-form expression can be obtained for predicting the unknown function $Y(\boldsymbol{x})$ at unobserved locations. Suppose that the function of interest Y is observed at the design setting $\{\boldsymbol{x}_i\}_{i=1}^n$; the observation vector then becomes $\boldsymbol{y} = [Y(\boldsymbol{x}_1), \ldots, Y(\boldsymbol{x}_n)]^T$. Having observed $\boldsymbol{y}$, the conditional mean of the process at a new point $\boldsymbol{x}_{\text{new}}$ is given by:

$$\hat{Y}(\boldsymbol{x}_{\text{new}}) = \mathbb{E}[Y(\boldsymbol{x}_{\text{new}})|\boldsymbol{y}] = \hat{\mu} + \boldsymbol{r}^T \boldsymbol{R}^{-1}(\boldsymbol{y} - \boldsymbol{1}_n \hat{\mu}) \tag{6}$$

where $\hat{\mu} = \boldsymbol{1}_n^T \boldsymbol{R}^{-1} y / \boldsymbol{1}_n^T \boldsymbol{R}^{-1} \boldsymbol{1}_n$ is the estimated value of μ, and $\boldsymbol{1}_n$ is an n-vector of 1's. Here, $\boldsymbol{R}$ is an $n \times n$ matrix whose (i, j)th entry is $R(\boldsymbol{x}_i, \boldsymbol{x}_j)$, and $\boldsymbol{r}$ is an n-vector whose ith entry is $R(\boldsymbol{x}_{\text{new}}, \boldsymbol{x}_i)$. A more detailed derivation of Eq. (6) can be found in [38,39].

While the predictor in Eq. (6) is easy to evaluate when the desired function Y is a scalar function, it becomes much more difficult to evaluate for the problem at hand, where the desired function involves spatiotemporal evolution. For example, in the present study of swirling flow dynamics, there are 380,928 grid points and 1000 time steps for each simulation case. Performing kriging for each grid point and time step would be impractical and time-consuming. From a statistical perspective, the use of separate kriging models over each grid point and time step also leads to a serious problem of over-parameterization (as each model requires d correlation parameters), which then results in poor prediction performance of the trained model. In light of these challenges and inspired by the fact that the numerical grid system remains the same for all simulated cases, we introduce an improved kriging-based model that combines the POD information from each design setting in the form of a "weighting parameter," named as *KSPOD*.

The key idea in KSPOD is to apply the kriging equation, Eq. (6), to predict the weight of each POD mode at observed points on a new design setting. To this end, the observations $\boldsymbol{y}$ are now taken to be the unit vector $\boldsymbol{e}_i$, where $\boldsymbol{e}_i$ is an n-vector with 1 in its ith element and 0 elsewhere. Intuitively, this quantifies the fact that the POD information extracted in the ith design setting corresponds to only that setting and not the other $n-1$ settings. With this in mind, the resulting predictor in Eq. (6) can then be viewed as the predicted weight for that particular POD term at a new design setting $\boldsymbol{x}_{\text{new}}$, denoted as $\hat{w}_{\text{new},i}$ This procedure is repeated for each of the n unit vectors $(\boldsymbol{e}_i)_{i=1}^n$, from which the n weighting parameter $(\hat{w}_{\text{new},i})_{i=1}^n$ can be obtained. They are subsequently used to predict the new POD modes and coefficients through a weighted average of the extracted modes and coefficients at the new design settings.

It is worth noting that the KSPOD methodology is an emulation technique, not a new POD method or another data decomposition method. The modes derived from KSPOD weighting parameters are not real POD modes, but a set of spatial functions (i.e., KSPOD spatial function) resulting from kriging. In other words, KSPOD uses a kriging-motivated weighing of modes for the goal of statistical prediction to approach the spatial functions for data reconstruction. The idea of weighting parameter is an accommodation that delivers efficient and effective estimation for the emulation model. Further details of weighting parameters are stated in the Appendix.

The algorithm in Table 3 outlines the detailed steps in the KSPOD algorithm. First, POD is performed for each simulated case to extract coherent structures. Next, the coefficients of POD modes are trained by ordinary kriging models using the Gaussian kernel in Eq. (5) as the correlation function, with the correlation parameter θ tuned using maximum-likelihood estimation, as implemented in the *R* package

Table 3 Algorithm of KSPOD

ALGORITHM: Kernel-smoothed POD		
Data:	For each design setting in $\{x_i\}_{i=1}^n$, provide the flow evaluation at each spatial location and time-step $f(x_i, u_j, t_q)$, where $\{u_j\}_{j=1}^J$ is the spatial location and $\{t_q\}_{q=1}^m$ is the time-step.	
Training:	*Step 1:*	For each design setting x_i, perform a POD and write it as $f(x_i, u_j, t_q) = \sum_{k=1}^K \beta^k(x_i, t_q)\phi^k(x_i, u_j)$.
	Step 2:	For each time-step t_q and each mode k, perform an ordinary kriging procedure on $\{\beta^k(x_1, t_q), \ldots, \beta^k(x_n, t_q)\}$ with inputs $\{x_1, \ldots, x_n\}$. The predictive function at an untried setting x_{new} is $\hat{\beta}^k(x_{\text{new}}, t_q)$.
	Step 3:	For $i = 1, \ldots, n$, perform an ordinary kriging procedure on e_i with inputs $\{x_1, \ldots, x_n\}$. The predictive function at an untried setting x_{new} is $\hat{w}_i(x_{\text{new}})$. Therefore, for each spatial location u_j and each mode k, the predictive KSPOD spatial function from $\phi^k(x_{\text{new}}, u_j)$ is $\hat{\phi}^k(x_{\text{new}}, u_j) = \sum_{i=1}^n \hat{w}_i(x_{\text{new}})\phi^k(x_i, u_j) / \sum_{i=1}^n \hat{w}_i(x_{\text{new}})$.
Prediction:	At an untried setting x_{new}, reconstruct $\hat{f}(x_{\text{new}}, u_j, t_q)$ by $\hat{f}(x_{\text{new}}, u_j, t_q) = \sum_{k=1}^K \hat{\beta}^k(x_{\text{new}}, t_q)\hat{\phi}^k(x_{\text{new}}, u_j)$, where $j = 1, \ldots, J$ and $q = 1, \ldots, m$.	

Table 4 Design matrix and associated inlet velocity information

Case	δ, mm	θ, deg	ΔL, mm	u_{in}, m/s	u_r, m/s	u_θ, m/s	K	Cluster
1	0.28	57.92	1.59	40.43	21.47	34.26	7.44	D
2	0.63	40.81	1.93	12.35	9.35	8.07	1.64	B
3	0.82	52.39	0.96	11.79	7.20	9.34	1.98	B
4	1.10	32.76	2.57	6.42	5.40	3.47	0.69	A
5	1.12	51.88	3.21	8.58	5.30	6.75	1.43	A
6	1.52	46.85	2.23	5.71	3.90	4.16	0.86	A
7	0.38	37.29	1.64	19.53	15.54	11.83	2.37	C
8	0.51	52.89	2.15	19.35	11.67	15.43	3.27	C
9	0.78	43.33	3.12	10.43	7.58	7.15	1.46	B
10	1.03	33.76	0.87	6.89	5.73	3.83	0.76	A
11	1.26	49.37	1.72	7.19	4.68	5.46	1.14	A
12	1.39	60.44	2.61	8.63	4.26	7.51	1.65	A
13	0.47	54.40	2.74	21.87	12.73	17.78	3.80	C
14	0.68	38.80	2.53	11.25	8.77	7.05	1.42	B
15	0.74	48.36	1.89	12.06	8.02	9.02	1.88	B
16	0.93	33.26	1.47	7.63	6.38	4.18	0.83	A
17	1.22	42.82	0.91	6.60	4.84	4.49	0.92	A
18	1.35	57.42	3.17	8.15	4.39	6.87	1.49	A
19	0.32	58.43	2.27	35.58	18.63	30.31	6.60	D
20	0.59	34.77	1.13	12.19	10.01	6.95	1.38	B
21	0.84	49.87	2.83	10.89	7.02	8.32	1.74	B
22	0.99	44.33	1.76	8.35	5.97	5.84	1.20	A
23	1.20	37.79	3.08	6.24	4.93	3.82	0.77	A
24	1.45	55.41	1.55	7.17	4.07	5.90	1.27	A
25	0.40	36.28	2.32	18.27	14.73	10.81	2.16	C
26	0.49	51.38	1.42	19.51	12.18	15.24	3.21	C
27	0.72	53.39	3.29	13.84	8.25	11.11	2.36	B
28	0.95	40.31	1.17	8.18	6.24	5.29	1.07	A
29	1.24	59.43	1.98	9.36	4.76	8.06	1.76	A
30	1.37	43.83	2.78	5.99	4.32	4.15	0.85	A

Note: Cases 1–6 are on slice 1, cases 7–12 on slice 2, cases 13–18 on slice 3, cases 19–24 on slice 4, and cases 25–30 on slice 5, corresponding to the symbols on Fig. 2.

"GPfit" [37]. The predictive function can be constructed based on Eq. (6). The weighting parameters are also trained using the procedure described above. Finally, the POD modes and coefficients are predicted, and are used to "reconstruct" the flowfield at the new design setting $\boldsymbol{x}_{\text{new}}$.

As in any physical or statistical model, there are implicit assumptions. First, by predicting the ith POD term of the new design setting using only the information for the ith POD terms extracted from observed design settings, we assume the ranking of the extracted flow physics from POD [corresponding to the rank of its corresponding expansion term in Eq. (2)] to be invariant over different geometry settings. In other words, the flow feature for the first POD mode corresponds to the same coherent structure over all n simulated design settings. The same holds true for the subsequent modes. Second, for the methodology to work, design settings whose dominant POD modes capture similar physics and dynamics should be clustered together for training. As information can be damped or even diminished during the training process, clustering the data and using cases with similar phase content can mitigate the problem.

The method is termed kernel-smoothed POD, because the kriging here does not apply the weighting parameter $\hat{w}_{\text{new},i}$ directly to the flowfield. The flow structures within the flowfield are a combination of waves with different frequencies, amplitudes, and phases. If the weighting parameter is used on the flowfield directly, the two datasets may cancel each other during the regression process, thereby losing useful information. The phase difference can be observed in POD modes as well. Application of weighting functions to POD modes with a kernel-smoothed algorithm can avoid the phase cancellation and retain important flow physics.

Once the KSPOD emulator is trained, it can be used to predict the flow evolution at a new design point. The computational cost is reduced by several orders of magnitude. For the demonstration case in the present study, high-fidelity simulations take around 100,000 CPU hours for each case. The trained model can evaluate a new flowfield with 1000 snapshots in 0.02 CPU hours.

III. Results and Discussion

A. High-Fidelity Simulation Results

The LES-based high-fidelity numerical framework described earlier was implemented for the 30 cases selected by SLHD for the three design parameters, which are decided by the sensitivity analysis using a first-order Monte Carlo estimation of Sobol indices [13]. The sensitivity analysis was performed based on the desired liquid-film thickness and spreading angle. The three chosen parameters are significantly influenced by the inlet velocity, u_{in}, which ranges from

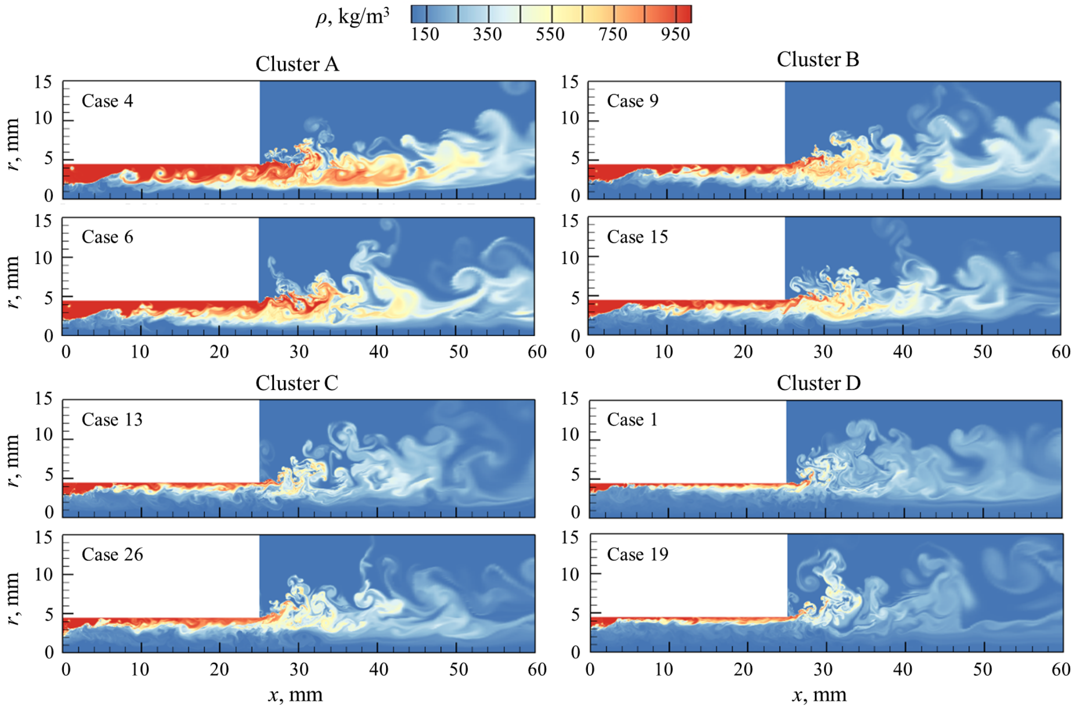

Fig. 3 Snapshots of density field in different clusters at $t = 1.21$ ms, obtained from LES-based high-fidelity simulations.

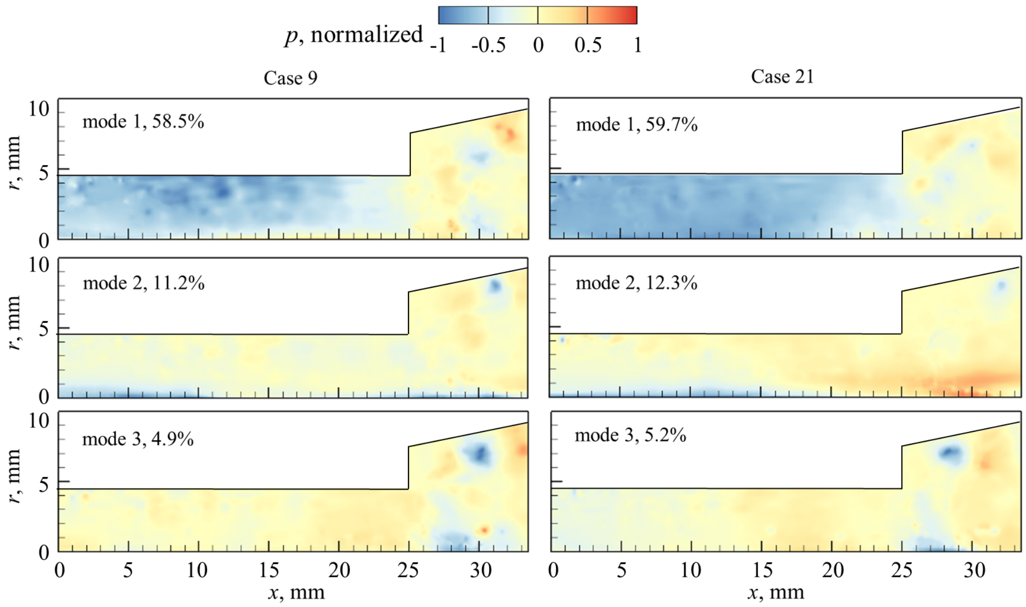

Fig. 4 Pressure POD modes 1, 2, and 3 for cases 9 and 21 from cluster B.

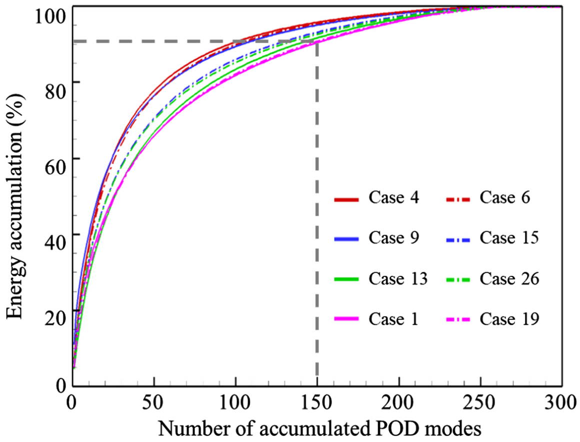

Fig. 5 Energy accumulation of density POD modes.

5.71 to 40.43 m/s as listed in Table 4. The 30 training cases are roughly classified into four groups in terms of u_{in} (m/s) as follows: cluster A with $u_{\mathrm{in}} < 10$; cluster B with $10 \le u_{\mathrm{in}} < 18$; cluster C with $18 \le u_{\mathrm{in}} < 25$; and cluster D with $u_{\mathrm{in}} > 25$.

Table 5 Design parameters for eight test cases in four different clusters

Case	δ, mm	θ, deg	ΔL, mm	u_{in}, m/s	u_r, m/s	u_θ, m/s	Cluster
A1	1.26	44.11	0.94	6.55	4.70	4.56	A
A2	1.20	41.97	0.90	6.65	4.94	4.44	A
B1	0.70	40.73	2.71	11.12	8.43	7.26	B
B2	0.71	52.59	3.24	13.79	8.38	10.95	B
C1	0.42	37.73	2.41	17.91	14.16	10.96	C
C2	0.49	57.12	2.88	22.33	12.12	18.75	C
D1	0.27	50.39	1.40	34.37	21.91	26.48	D
D2	0.33	60.76	2.32	36.32	17.74	31.70	D

Overall, there are 15 cases in cluster A, 8 in cluster B, 5 in cluster C, and 2 in cluster D. Figure 3 shows two sample snapshots of the density field for each cluster. Variations of film thickness and spreading angle are observed. Cluster A has the slowest inlet velocity and generates thicker film and smaller spreading angle. The film thickness decreases and the spreading angle increases between clusters B and C; the smallest film thickness and largest spreading angle appear in cluster D, where there is the highest inlet velocity.

As mentioned earlier, the geometric constant K is an indicator of swirl strength, with a higher value implying stronger azimuthal

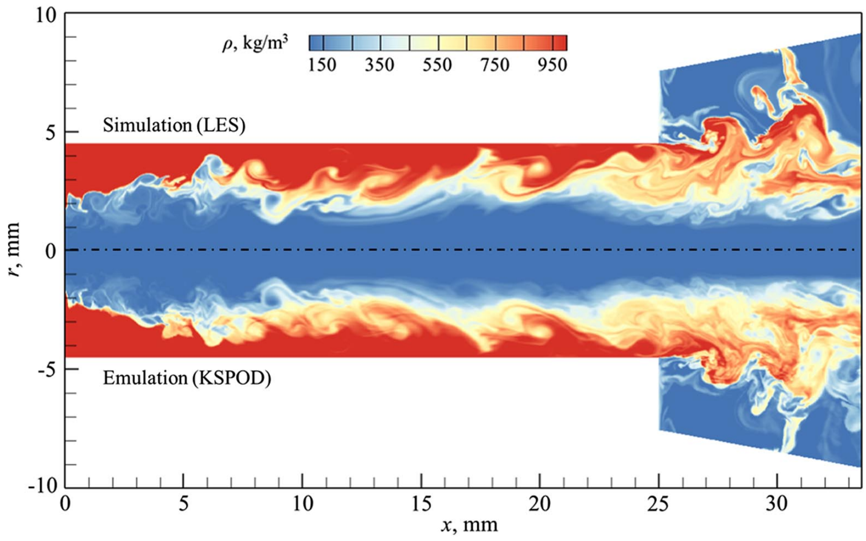

Fig. 6 Comparison of density field between LES-based simulation and prediction by KSPOD-based emulation. Test case A1.

Table 6 Weighting parameters from of POD modes for case A1

Case	1	2	3	4	5	6	7	8	9	10
Cluster	D	B	B	A	A	A	C	C	B	A
Weighting parameter	−0.01%	0.50%	2.13%	0.82%	−4.24%	0.02%	0.00%	0.12%	0.04%	−2.18%
Case	11	12	13	14	15	16	17	18	19	20
Cluster	A	A	C	B	B	A	A	A	D	B
Weighting parameter	2.17%	−0.03%	0.16%	0.18%	−0.13%	0.00%	91.77%	−0.17%	−0.04%	−0.08%
Case	21	22	23	24	25	26	27	28	29	30
Cluster	B	A	A	A	C	C	B	A	A	A
Weighting parameter	0.16%	−2.50%	0.20%	−0.13%	0.05%	0.01%	0.00%	−2.28%	0.08%	13.37%

momentum. When K becomes small, the swirling flow does not contain enough azimuthal momentum to produce a large spreading angle of the liquid film. This suggests that K plays a pivotal role, with a slight change causing significant differences in flow dynamics. Therefore, the training process must preserve important flow physics during data reduction.

B. Prediction by KSPOD-Based Emulation

Figure 4 compares the first three pressure POD modes between cases 9 and 21 in cluster B. Mode 1 is an ensemble of the longitudinal modes of the hydrodynamic instability. In both cases, the injector length is fixed at 25 mm, and the longitudinal hydrodynamic wave speed in the liquid film is estimated to be 7–9 m/s [14–16,40]. This

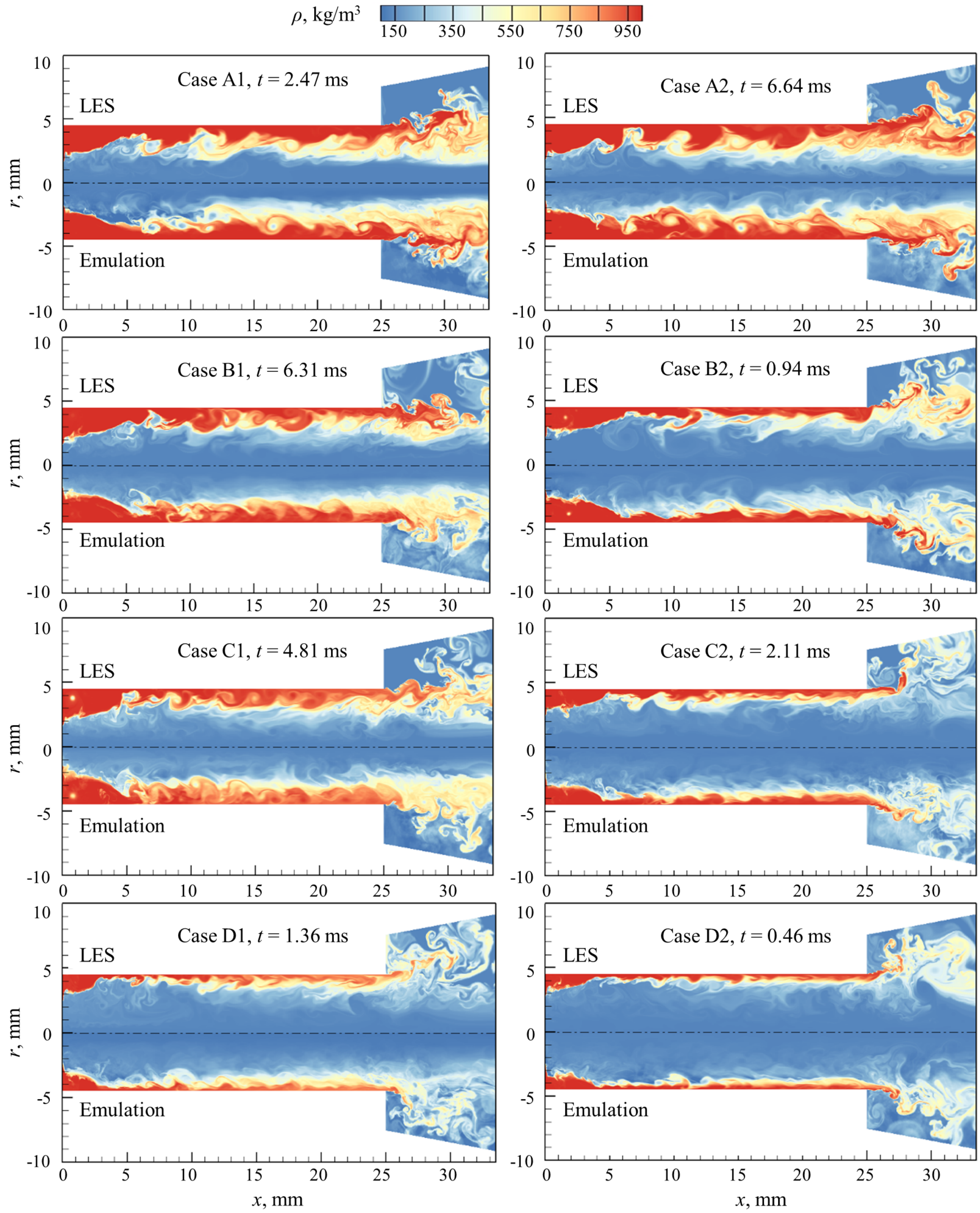

Fig. 7 Comparison of density field between LES-based simulation and prediction by KSPOD-based emulation.

Table 7 Film thickness and spreading angle for simulation and emulation results

	Spreading angle			Film thickness		
Case	Simulation	Emulation	$\overline{\varepsilon}_{\mathrm{abs}}$, %	Simulation	Emulation	$\overline{\varepsilon}_{\mathrm{abs}}$, %
A1	52.85	52.92	0.14%	0.629	0.625	0.51%
A2	52.57	51.96	1.15%	0.637	0.657	3.14%
B1	54.22	53.66	1.02%	0.582	0.600	3.03%
B2	53.81	53.87	0.12%	0.594	0.592	0.40%
C1	57.68	57.71	0.05%	0.469	0.473	0.85%
C2	57.78	57.74	0.06%	0.475	0.472	0.63%
D1	59.00	58.03	1.64%	0.379	0.378	0.26%
D2	61.59	61.33	0.41%	0.370	0.377	1.97%

leads to a characteristic frequency of 0.32–0.45 kHz for the hydrodynamic instability. Mode 2 contains about 12% of the total energy and has similar structures similar cases 9 and 21, except in the downstream region close to the centerline. Mode 3 has around 5% of the total energy, and the distributions for the two cases are alike. The dominant frequencies for mode 2 in the two cases are 0.61 and 0.63 kHz, respectively. For mode 3, dominant frequencies of 1.14 and 1.39 kHz are observed. The frequencies associated with the recirculating flow downstream of the injector exit are 0.65 and 1.29 kHz, respectively. Hence, modes 2 and 3 can be attributed to excitation by the processing vortex core. In short, modes 1–3 capture similar physical characteristics in the dominant POD modes with the same order resulting from the eigen-decomposition. The similarity of POD modes among cases in the same cluster justifies the two fundamental assumptions of KSPOD stated on Sec. I.

Figure 5 shows the accumulated energy percentage of the density POD modes for the cases presented in Fig. 3. The first 150 modes contain over 90% of the total energy, and the first 300 modes have over 99.9% of the energy. The latter is sufficient to cover the energy spectrum of the density field and thus selected for the KSPOD analysis. KSPOD builds a posterior model based on the basis functions ranked by the eigen-decomposition. As such, the reduced data sets (i.e., the dominant modes) capture the significant flow structures and their dynamic characteristics for the same rank.

The 30 cases selected by DoE have different inlet velocities, which generate different instability wave speeds. Phase information, however, cannot be reproduced perfectly, due to the presence of turbulence. If two designs with similar dynamic mechanisms, such as cases 4 and 6, have phase differences of 90° or 180°, the prediction results may be excessively smoothed by traditional kriging. If the weighting parameters evaluated by kriging for these two cases are close, and their instability waves are exactly out of phase, the wave information cancels out. To avoid this situation and ensure the two main assumptions proposed in this paper viable, the POD signs have to be manually checked. In the current study, there are only few cases that require to be adjusted manually. Once the sign of a POD mode is changed, the corresponding coefficient must be modified as well. This manual work is performed before the kriging process to ensure that no wave information will be canceled out after applying the weighting parameters. One of our future objectives is to improve the emulation technique without manually changing the sign of POD modes and coefficients. As such, KSPOD applies a weighting function for the POD modes to ensure that similar POD modes in different cases can retain appropriate phase information.

Figure 6 shows excellent comparison between the simulation result and prediction from the trained emulator for test case A1. The design parameters are given in Table 5. The KSPOD-based emulator works well; it is able to capture essential flow structures successfully. The evolution of the liquid film and its spreading downstream of the injector exit also agree well between the simulation and emulation. The overall turnaround time for emulation prediction, excluding data loading and training, is about 42 s of CPU time for one snapshot. Table 6 lists the contribution of POD modes from each case. The weighting parameter is calculated based on the test case A1. It is noted that cases 17 and 30 provide, respectively, 91.77% and 13.37% weighting, whereas cases 5, 10, 22, and 28 provide over 10% negative weights on A1.

Figure 7 shows prediction results for the four different design clusters, in each of which two test cases are considered. The flow structures and dynamics are well captured by the emulator. In case C1, the corner recirculation near the headend of the injector is clearly observed. The discrepancy caused by time delay is present in cases A1 and B2. In A1, the traveling surface wave in the injector propagates downstream slightly faster in the simulation than in the emulation.

C. Analysis of Prediction with Performance Measures: Film Thickness and Spreading Angle

To further evaluate the accuracy of the KSPOD-based emulation, two performance measures, film thickness and spreading angle, are assessed. Table 7 presents a comparison of the time-mean simulation and emulation (prediction) results, obtained by averaging the instantaneous data over a statistically meaningful time duration. The error is calculated as follows.

$$\varepsilon_{\mathrm{abs}} = \frac{|x_{\mathrm{sim}} - x_{\mathrm{emu}}|}{x_{\mathrm{sim}}} \times 100\% \tag{7}$$

where x_{sim} represents data from simulation and x_{emu} from emulation. As most training cases are located in clusters A and B, the predictions are more robust within these two clusters.

The probability densities of instantaneous spreading angle and liquid film thickness for four selected training cases in cluster A are obtained from the estimated kernel smoothing function, as shown in Fig. 8. A kernel distribution is a nonparametric representation of the probability density function, $f_h(x)$, of a random variable, written as

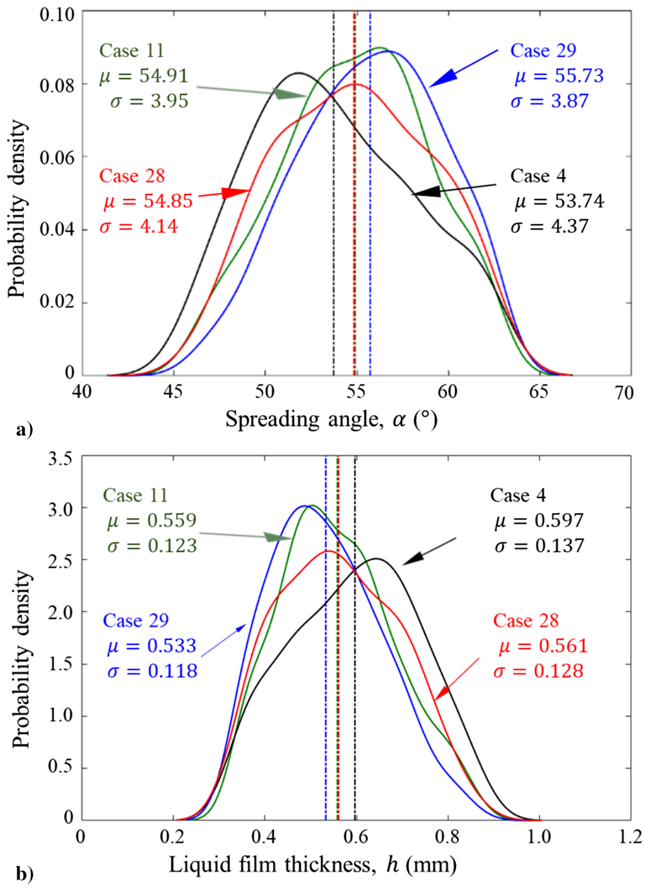

Fig. 8 Probability densities of instantaneous spreading angle and liquid film thickness for cases 4, 11, 28, and 29 from cluster A. Vertical lines represent mean values.

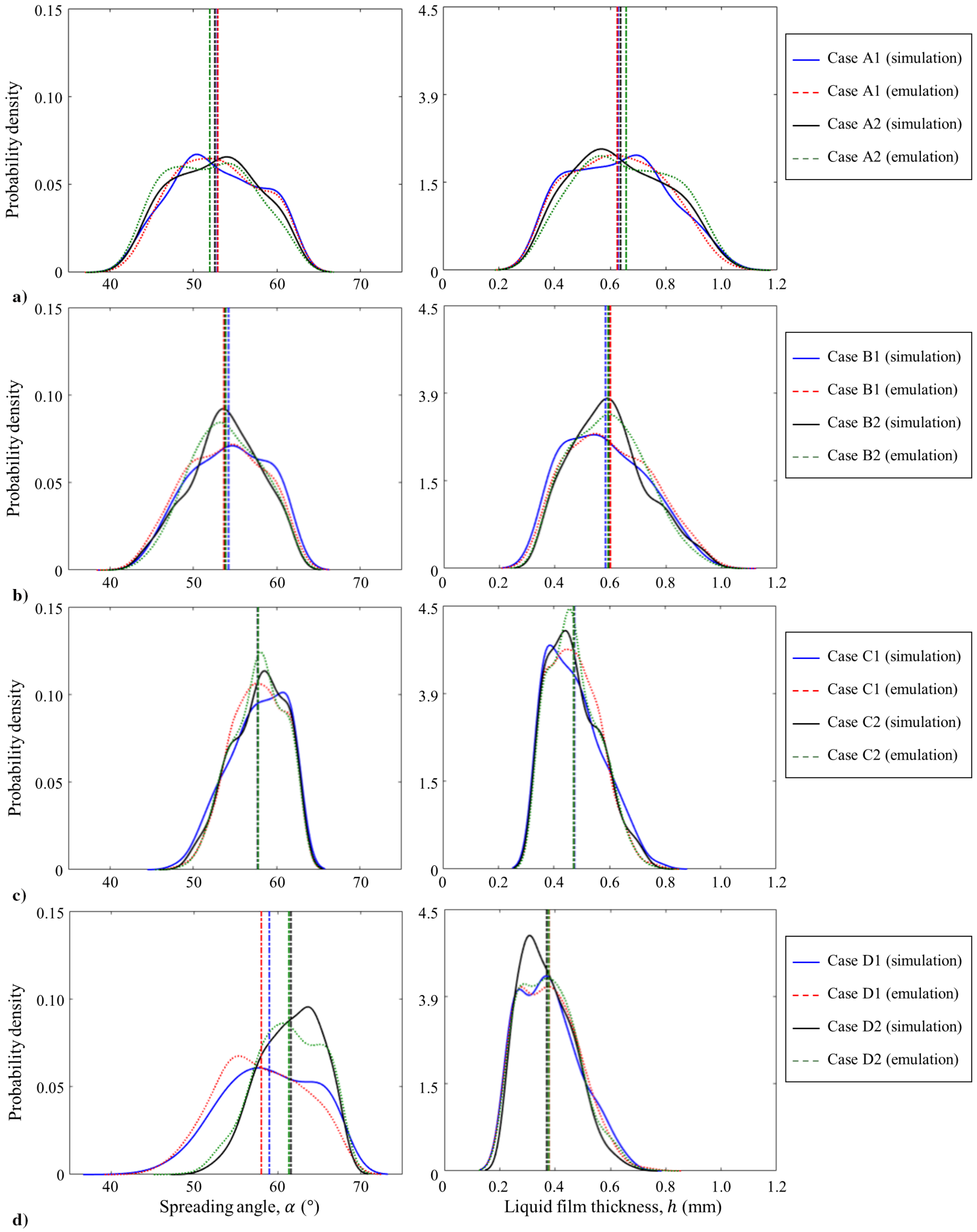

Fig. 9 Probability densities of instantaneous spreading angle and liquid film thickness for test cases. Vertical lines represent mean values.

$$f_h(\boldsymbol{x}) = \frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x - x_i}{h}\right) \tag{8}$$

where n is the sample size, $\mathcal{K}(\cdot)$ the density smoothing function, and h a smoothing parameter named bandwidth. The distributions of probability density for cases in the same cluster bear close similarities, but do not collapse.

Figure 9 compares the probability density distributions of spreading angle and liquid film thickness between simulations and emulations for all test cases. The vertical lines represent mean values. Detailed information about mean values, standard deviations, and averaged absolute error $\overline{\varepsilon_{\text{abs}}}$ is given in Table 7. The maximum absolute errors for liquid film thickness and spreading angle are 1.64% in case D1 and 3.14% in case A2, respectively. The liquid film thicknesses are relatively small in clusters C and D, and so their mean values visually overlap each other.

Another way to measure the performance of the emulation is based on the distribution of liquid film thickness along the axial direction. Figure 10 shows a comparison between the simulation and emulation results, averaged over 1000 snapshots. Figure 11 shows the absolute error between simulations and emulations. The horizontal lines represent averaged absolute error $\overline{\varepsilon_{\text{abs}}}$ for each test case. Overall, the liquid film thickness predicted by the KSPOD-based emulation has an averaged error less than 5%, except for case B2, which has $\overline{\varepsilon_{\text{abs}}} = 6.4\%$. Cases with higher inlet velocities have less variation for

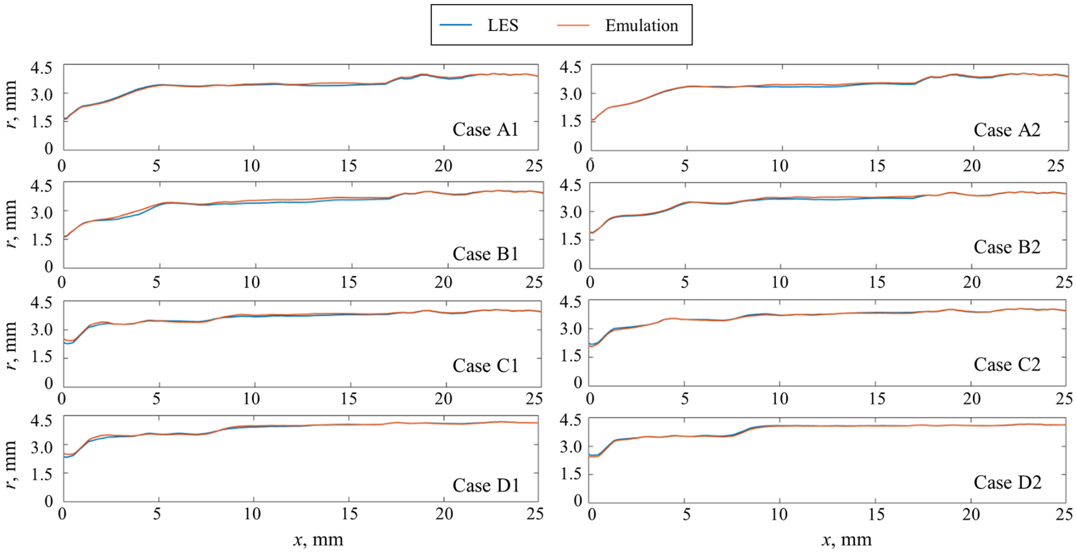

Fig. 10 Comparison of liquid film thickness along the axial direction, averaged over 1000 snapshots.

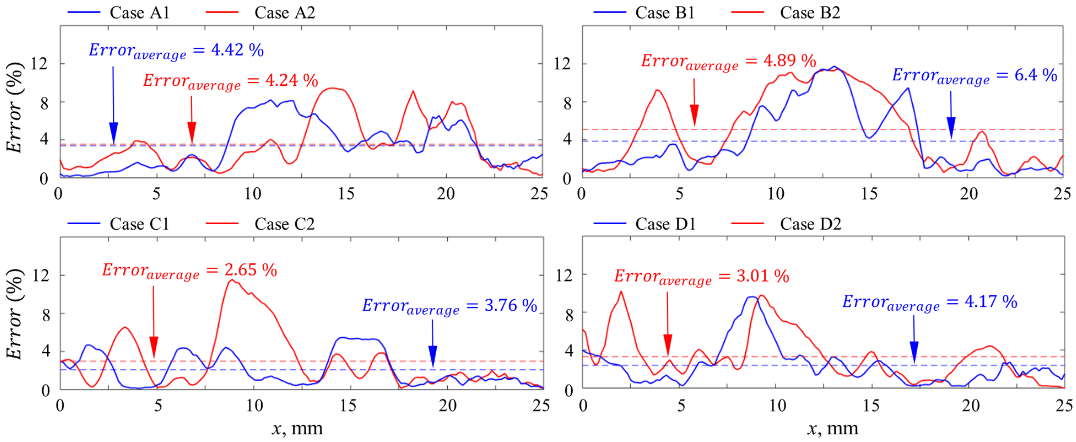

Fig. 11 Error for liquid film thickness along the axial direction.

the liquid film thickness near the injector exit. The first local maximum of error occurs in the LOX inlet area, the region that contains the highest momentum and kinetic energy. The second peak of error takes place when the flowfield is still developing. The large error along the axial direction in case B2 can be attributed to the fact that this case is close to the boundary between clusters B and C, and includes more flow mechanisms that are prominent in the cases in cluster C. The 30 cases used for model training are mostly distributed in clusters A and B. If the 30 training cases had space-filling properties with respect to inlet velocities, the prediction error could be decreased and the quantitative analysis for test cases B2, C1, and C2 could be improved.

IV. Conclusions

This study develops statistical methods for building an efficient and reliable emulation (surrogate) model for the prediction of spatiotemporal flow dynamics. The emulation method is constructed on a kernel-smoothed POD (KSPOD) technique, which leverages kriging-based weighted functions from the design matrix. As an example, the spatiotemporal flow evolution in a swirl injector is investigated over a wide range of design parameters and operating conditions. The KSPOD-based emulation model is validated against high-fidelity simulation results obtained from large-eddy simulation. The model not only preserves key physical mechanisms underlying the flow evolution, but also quantitatively captures the dynamics over a wide range of temporal and spatial scales of concern. The KSPOD-based emulation model has successfully reproduced some flow dynamics information, including liquid film thickness and spreading angle with good accuracy (most errors are under 2%). Qualitatively, the KSPOD prediction flow fields perform nice visualization results with detailed vortex structures. In addition, the model enables effective design surveys using high-fidelity simulation data, achieving a turnaround time for evaluating new design points that is 42,000 times faster than the original simulation.

Based on the current results, further investigation is necessary to improve the emulation model, data reduction, and feature extraction of flow dynamics. The KSPOD could also be combined with common proper-orthogonal-decomposition techniques to mitigate the uncertainties of prediction for a broad-range design space. Future works will aim at further decreasing uncertainties and errors of the surrogate model. Moreover, the use of artificial neural networks could potentially be employed if an activation function that properly treats the physics of the problem can be identified. More effective incorporation of physical knowledge for model tuning should also be explored and implemented.

Appendix: Interpretation as a Nadaraya-Watson Kernel Smoother

Suppose that simulations are conducted at various design geometries $\boldsymbol{c} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n\}$, and it is assumed that the true function $W(\boldsymbol{c})$ is a realization from a stochastic process

$$W(\boldsymbol{c}) = \mu + Z(\boldsymbol{c}) \tag{A1}$$

where $\boldsymbol{c}$ is an n-dimentional vector (with d design variables), μ a constant global model, and $Z(\boldsymbol{c})$ a local deviation from the global model with zero mean. Consider the kriging of the indicator vector $\boldsymbol{e}_i$, an n-vector with unity in the ith entry, and zero elsewhere. There is another vector $\boldsymbol{r}_i$, where $i = 1, \ldots, p$ and p is the number of control settings for design cases. Because a space-filling design is employed, it is easy to show that the optimal correlation parameters for the underlying Gaussian process should be equal for all p dimensions. Denote this common correlation as θ. When the number of design points $n \to \infty$, one can show that $\theta \to \infty$ as well, because the "kriging surface" for $\boldsymbol{e}_i$ converges pointwise to a discontinuous surface with value 1 at $\boldsymbol{r}_i$ and 0 elsewhere. The kriging estimate $\boldsymbol{e}_i$ for a new design setting $\boldsymbol{r}_{\text{new}}$ is

$$\hat{w}_{\text{new},i} = \hat{\mu} + \boldsymbol{r}_{\text{new}}^T \boldsymbol{R}^{-1}(\boldsymbol{e}_i - \hat{\mu}\mathbf{1}_n) \tag{A2}$$

where $\hat{\mu} = (\mathbf{1}_n^T \boldsymbol{R}^{-1}\mathbf{1}_n)^{-1}\mathbf{1}_n^T \boldsymbol{R}^{-1}$, $\boldsymbol{e}_i = \boldsymbol{R}_{ii}^{-1} / \sum_{i=1}^{n} \boldsymbol{R}_{ii}^{-1}$, $\boldsymbol{R} \equiv \text{Corr}[Z(\boldsymbol{c}_i), Z(\boldsymbol{c}_j)]$ are $n \times n$ correlation matrices, and $\hat{w}_{\text{new},i}$ a weighted number based on kriging. When $\theta \to \infty$ and $n \to \infty$, the best linear unbiased predictor estimator $\hat{\mu} \to 0$; when $\theta \to \infty$, the inverse correlation matrix $\boldsymbol{R}^{-1}$ converges elementwise to $\boldsymbol{I}_n$. Under these two approximations, we get a new kernel

$$k_\theta(\boldsymbol{c}_i, \boldsymbol{c}_{\text{new}}) = \hat{w}_{\text{new},i} \approx \boldsymbol{r}_{\text{new}}^T \boldsymbol{R}^{-1}\boldsymbol{e}_i = \exp\{-\theta\|\boldsymbol{c}_i - \boldsymbol{c}_{\text{new}}\|_2^2\} \tag{A3}$$

where $\|\cdot\|_2$ is the Euclidean norm. In other words, $\hat{w}_{\text{new},i}$ is the isotropic Gaussian kernel $k_\theta(\boldsymbol{c}_i, \boldsymbol{c}_{\text{new}})$. The proposed predictor of the first kth mode at the new design setting $\boldsymbol{c}_{\text{new}}$ is

$$\hat{\phi}_{\text{new}}^k(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} \hat{w}_{\text{new},i}\phi_i^k(\boldsymbol{x})}{\sum_{i=1}^{n} \hat{w}_{\text{new},i}} \approx \frac{\sum_{i=1}^{n} k_\theta(\boldsymbol{c}_i, \boldsymbol{c}_{\text{new}})\phi_i^k(\boldsymbol{x})}{\sum_{i=1}^{n} k_\theta(\boldsymbol{c}_i, \boldsymbol{c}_{\text{new}})} \tag{A4}$$

where $\phi_i^k(\boldsymbol{x})$, $i = 1, \ldots, n$—the kth POD mode at design setting $\boldsymbol{c}_i$.

With kriging over k modes, the new POD coefficient is defined as

$$\hat{\beta}_{\text{new}}^k(x, t) = (\hat{\beta}_1^k, \hat{\beta}_2^k, \ldots, \hat{\beta}_{10d}^k) \tag{A5}$$

and the corresponding POD expansion using kth modes (i.e., the prediction) is given by

$$\begin{aligned} \hat{f}_{\text{new}}(x, t) &= \sum_{k=1}^{\infty} \hat{\beta}_{\text{new}}^k(x, t)\hat{\phi}_{\text{new}}^k(x) \\ &= \sum_{k=1}^{\infty} \frac{\beta_{\text{new}}^k(x, t)(\sum_{i=1}^{n} k_\theta(\boldsymbol{c}_i, \boldsymbol{c}_{\text{new}})\phi_i^k(x))}{(\sum_{i=1}^{n} k_\theta(\boldsymbol{c}_i, \boldsymbol{c}_{\text{new}}))^2} \end{aligned} \tag{A6}$$

Equation (A3) can be seen as a kernel smoother on the observed modes $\{\phi_i^k(\boldsymbol{x})\}_{i=1}^n$ and coefficients $\{\beta_i^k(\boldsymbol{x}, t)\}_{i=1}^n$.

Acknowledgments

This work was sponsored partly by the William R. T. Oakes Endowment and partly by the Coca-Cola Endowment of the Georgia Institute of Technology.

References

[1] Ling, J., and Templeton, J., "Evaluation of Machine Learning Algorithms for Prediction of Regions of High Reynolds Averaged Navier Stokes Uncertainty," *Physics of Fluids*, Vol. 27, No. 8, 2015, Paper 085103.
doi:10.1063/1.4927765

[2] Ling, J., Ruiz, A., Lacaze, G., and Oefelein, J., "Uncertainty Analysis and Data-Driven Model Advances for a Jet-in-Crossflow," *Journal of Turbomachinery*, Vol. 139, No. 2, 2017, Paper 021008.
doi:10.1115/1.4034556

[3] Parish, E. J., and Duraisamy, K., "A Paradigm for Data-Driven Predictive Modeling Using Field Inversion and Machine Learning," *Journal of Computational Physics*, Vol. 305, Jan. 2016, pp. 758–774.
doi:10.1016/j.jcp.2015.11.012

[4] Singh, A. P., Medida, S., and Duraisamy, K., "Machine-Learning-Augmented Predictive Modeling of Turbulent Separated Flows over Airfoils," *AIAA Journal*, Vol. 55, No. 7, 2017, pp. 2215–2227.
doi:10.2514/1.J055595

[5] Yang, V., "Modeling of Supercritical Vaporization, Mixing, and Combustion Processes in Liquid-Fueled Propulsion Systems," *Proceedings of the Combustion Institute*, Vol. 28, No. 1, Jan. 2000, pp. 925–942.
doi:10.1016/S0082-0784(00)80299-4

[6] Zong, N., and Yang, V., "Cryogenic Fluid Jets and Mixing Layers in Transcritical and Supercritical Environments," *Combustion Science and Technology*, Vol. 178, Nos. 1–3, Jan. 2006, pp. 193–227.
doi:10.1080/00102200500287613

[7] Huo, H., and Yang, V., "Large-Eddy Simulation of Supercritical Combustion: Model Validation Against Gaseous H_2O_2 Injector," *Journal of Propulsion and Power*, Vol. 33, No. 5, 2017, pp. 1272–1284.
doi:10.2514/1.B36368

[8] Wang, X., Wang, Y., and Yang, V., "Geometric Effects on Liquid Oxygen/Kerosene Bi-Swirl Injector Flow Dynamics at Supercritical Conditions," *AIAA Journal*, Vol. 55 No. 10, 2017, pp. 3467–3475.
doi:10.2514/1.J055952

[9] Wang, X., Huo, H., Unnikrishnan, U., and Yang, V., "A Systematic Approach Toward High-Fidelity Modeling and Efficient Simulation of Supercritical Fluid Mixing and Combustion," *Combustion and Flame*, Vol. 195, Sept. 2018, pp. 203–215.
doi:10.1016/j.combustflame.2018.04.030

[10] Yeh, S.-T., Wang, X., Sung, C.-L., Mak, S., Chang, Y.-H., Zhang, L., Wu, C.-F. J., and Yang, V., "Common Proper Orthogonal Decomposition-Based Spatio-Temporal Emulator for Design Exploration," *AIAA Journal*, Vol. 56, No. 6, 2018, pp. 2429–2442.
doi:10.2514/1.J056640

[11] Wang, X., Yeh, S.-T., Chang, Y.-H., and Yang, V., "A High-Fidelity Design Methodology Using LES-Based Simulation and POD-Based Emulation: A Case Study of Swirl Injectors," *Chinese Journal of Aeronautics*, Vol. 31, No. 9, 2018, pp. 1855–1869.
doi:10.1016/j.cja.2018.07.004

[12] Mak, S., Sung, C.-L., Wnag, X., Yeh, S.-T., Chang, Y.-H., Joseph, V. R., Yang, V., and Wu, C.-F. J., "An Efficient Surrogate Model for Emulation and Physics Extraction of Large Eddy Simulations," *Journal of the American Statistical Association*, Vol. 113, No. 524, 2018, pp. 1443–1456.
doi:10.1080/01621459.2017.1409123

[13] Bazarov, V. G., and Yang, V., "Liquid-Propellant Rocket Engine Injector Dynamics," *Journal of Propulsion and Power*, Vol. 14, No. 5, 1998, pp. 797–806.
doi:10.2514/2.5343

[14] Zong, N., and Yang, V., "Cryogenic Fluid Dynamics of Pressure Swirl Injectors at Supercritical Conditions," *Physics of Fluids*, Vo. 20, No. 5, 2008, Paper 056103.
doi:10.1063/1.2905287

[15] Wang, X., Huo, H., Wang, Y., and Yang, V., "Comprehensive Study of Cryogenic Fluid Dynamics of Swirl Injectors at Supercritical Conditions," *AIAA Journal*, Vol. 55, No. 9, 2017, pp. 3109–3119.
doi:10.2514/1.J055868

[16] Loeppky, J. L., Sacks, J., and Welch, W. J., "Choosing the Sample Size of a Computer Experiment: A Practical Guide," *Technometrics*, Vol. 51, No. 4, 2009, pp. 366–376.
doi:10.1198/TECH.2009.08040

[17] Benner, P., Gugercin, S., and Willcox, K., "A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems," *SIAM Review*, Vol. 57, No. 4, 2015, pp. 483–531.
doi:10.1137/130932715

[18] Cressie, N., "The Origins of Kriging," *Mathematical Geology*, Vol. 22, No. 3, 1990, pp. 239–252.
doi:10.1007/BF00889887

[19] Berkooz, G., Holmes, P., and Lumley, J. L., "The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows," *Annual Review of Fluid Mechanics*, Vol. 25, No. 1, 1993, pp. 539–575.
doi:10.1146/annurev.fl.25.010193.002543

[20] Chen, X., and Yang, V., "Effect of Ambient Pressure on Liquid Swirl Injector Flow Dynamics," *Physics of Fluids*, Vol. 26, No. 10, 2014, Paper 102104.
doi:10.1063/1.4899261

[21] Joseph, V. R., Gul, E., and Ba, S., "Maximum Projection Designs for Computer Experiments," *Biometrika*, Vol. 102, No. 2, 2015, pp. 371–380.
doi:10.1093/biomet/asv002

[22] Meng, H., and Yang, V., "A Unified Treatment of General Fluid Thermodynamics and Its Application to a Preconditioning Scheme," *Journal of Computational Physics*, Vol. 189, No. 1, 2003, pp. 277–304.
doi:10.1016/S0021-9991(03)00211-0

[23] Zong, N., and Yang, V., "An Efficient Preconditioning Scheme for real-Fluid Mixtures Using Primitive Pressure–Temperature Variables," *International Journal of Computational Fluid Dynamics*, Vol. 21, Nos. 5–6, 2007, pp. 217–230.
doi:10.1080/10618560701584373

[24] Hsieh, S. Y., and Yang, V., "A Preconditioned Flux-Differencing Scheme for Chemically Reacting Flows at All Mach Numbers," *International Journal of Computational Fluid Dynamics*, Vol. 8, No. 1, 1997, pp. 31–49.
doi:10.1080/10618569708940794

[25] Li, H. G., Zong, N., Lu, X. Y., and Yang, V., "A Consistent Characteristic Boundary Condition for General Fluid Mixture and Its Implementation in a Preconditioning Scheme," *Advances in Applied Mathematics and Mechanics*, Vol. 4, No. 1, 2012, pp. 72–92.
doi:10.4208/aamm.11-m1160

[26] Mani, A., "Analysis and Optimization of Numerical Sponge Layers as a Nonreflective Boundary Treatment," *Journal of Computational Physics*, Vol. 231, No. 2, 2012, pp. 704–716.
doi:10.1016/j.jcp.2011.10.017

[27] Oefelein, J. C., and Yang, V., "Modeling High-Pressure Mixing and Combustion Processes in Liquid Rocket Engines," *Journal of Propulsion and Power*, Vol. 14, No. 5, 1998, pp. 843–857.
doi:10.2514/2.5349

[28] Meng, H., and Yang, V., "A Unified Treatment of General Fluid Thermodynamics and Its Application to a Preconditioning Scheme," *Journal of Computational Physics*, Vol. 189, No. 1, 2003, pp. 277–304.
doi:10.1016/S0021-9991(03)00211-0

[29] Zong, N., Meng, H., Hsieh, S. Y., and Yang, V., "A Numerical Study of Cryogenic Fluid Injection and Mixing Under Supercritical Conditions," *Physics of Fluids*, Vol. 16, No. 12, 2004, pp. 4248–4261.
doi:10.1063/1.1795011

[30] Meng, H., Hsiao, G. C., Yang, V., and Shuen, J. S., "Transport and Dynamics of Liquid Oxygen Droplets in Supercritical Hydrogen Streams," *Journal of Fluid Mechanics*, Vol. 527, March 2005, pp. 115–139.
doi:10.1017/S0022112004003106

[31] Xiong, F., Xiong, Y., Chen, W., and Yang, S., "Optimizing Latin Hypercube Design for Sequential Sampling of Computer Experiments," *Engineering Optimization*, Vol. 41, No. 8, 2009, pp. 793–810.
doi:10.1080/03052150902852999

[32] Qian, P. Z., "Sliced Latin Hypercube Designs," *Journal of the American Statistical Association*, Vol. 107, No. 497, 2012, pp. 393–399.
doi:10.1080/01621459.2011.644132

[33] Ba, S., Myers, W. R., and Brenneman, W. A., "Optimal Sliced Latin Hypercube Designs," *Technometrics*, Vol. 57, No. 4, 2015, pp. 479–487.
doi:10.1080/00401706.2014.957867

[34] Huang, Y., Wang, S., and Yang, V., "Systematic Analysis of Lean-Premixed Swirl-Stabilized Combustion," *AIAA Journal*, Vol. 44, No. 4, 2006, pp. 724–740.
doi:10.2514/1.15382

[35] Wackernagel, H., *Multivariate Geostatistics: An Introduction with Applications*, Springer, Berlin, 2003.
doi:10.1007/978-3-662-05294-5

[36] Joseph, V. R., Hung, Y., and Sudjianto, A., "Blind Kriging: A New Method for Developing Metamodels," *Journal of Mechanical Design*, Vol. 130, No. 3, 2008, Paper 031102.
doi:10.1115/1.2829873

[37] MacDonald, B., Ranjan, P., and Chipman, H., "GPfit: An R Package for Fitting a Gaussian Process Model to Deterministic Simulator Outputs," *Journal of Statistical Software*, Vol. 64, No. 1, 2015, pp. 1–23.
doi:10.18637/jss.v064.i12

[38] Santner, T. J., Williams, B. J., and Notz, W. I., *The Design and Analysis of Computer Experiments*, Springer, New York, 2003.
doi:10.1007/978-1-4757-3799-8

[39] Koehler, J. R., and Owen, A. B., "9 Computer Experiments," *Handbook of Statistics*, Vol. 13, 1996, pp. 261–308.
doi:10.1016/S0169-7161(96)13011-X

[40] Bazarov, V., Yang, V., and Puri, P., "Design and Dynamics of Jet and Swirl Injectors," *Liquid Rocket Thrust Chambers: Aspects of Modeling, Analysis, and Design*, Vol. 200, 2004, pp. 19–103.
doi:10.2514/5.9781600866760.0019.0103

C. W. Rowley
Associate Editor