# Introduction to A Probabilistic Approach Towards Uncertainty In Computations of Airborne Ash Clouds, using BENT and PUFF

Puneet Singla* Reza Madankan† Abani Patra‡ Jose L. Palma Lizana§

$19^{th}$ May, 2011

## 1 Volcanic Plumes

Particle transport models can be divided into two broad categories: those intended to calculate eruption column characteristics based on tephra fall deposits, as in [1], and those intended to predict long-range atmospheric and deposit distributions based on the scale of the eruption, as in [2]. Both types of models rely on the existence of an explicit relationship between the eruption and atmospheric dynamics and the resulting fall deposit. This relationship is complicated by a number of factors, including plume mechanics, variable weather conditions, and particle re-entrainment. Our interest is in the movement of ash clouds, and not in tephra deposition, and therefore we focus attention on long-range modeling. Thus we will ignore detailed topographical features and low atmosphere phenomena that is important for particle fallout and deposition. Instead we consider a simple particle transport model, but one that nonetheless contains several sources of uncertainty.

Tanaka [3] and Searcy et al. [4] developed PUFF, an ash tracking model for predicting the paths of young volcanic clouds. PUFF simplifies the eruption plume to a vertical source, and uses a Lagrangian pseudo-particle representation of the ash cloud in a detailed 3-D regional windfield to determine the trajectory of the cloud. PUFF and other dispersion models have proven extremely useful in modeling the distal transport of ash for aviation safety [4].

### 1.1 The PUFF Simulation Model

There are several Volcanic Ash Transport and Dispersion (VATD) models that have been used extensively for ash cloud forecasting and post-event analysis [5, 6, 7].

Unlike other VATD models, it is designed solely for predictions of ash-cloud movement in space and time, and it generates displays that can be tailored to user needs at volcano observatories, in a real-time setting during an eruption crisis. For these reasons, PUFF is necessarily simple by design, but it can be intricate in its implementation. The model takes into account the predominant physical processes that control particle movement, such as, winds, dispersion and settling, but it does not account for small scale physical processes that play a lesser role. In some cases we do not

*Assistant Professor, Department of Mechanical & Aerospace Engineering, University at Buffalo, Buffalo, NY-14260, Email: psingla@buffalo.edu.

†Graduate Student, Department of Mechanical & Aerospace Engineering, University at Buffalo, Buffalo, NY-14260, Email: rm93@buffalo.edu.

‡Professor, Department of Mechanical & Aerospace Engineering, University at Buffalo, Buffalo, NY-14260, Email: abani@buffalo.edu.

§Postdoctoral Associate, Geology, University at Buffalo, Buffalo, NY-14260, Email: josepalm@buffalo.edu.

understand in detail how these sub-scale process impacts the overall cloud movement (e.g. inter-particle interactions), in other cases the required information may not be available in a real-time situation (e.g. precipitation), and in yet other cases inclusion of these processes may significantly lengthen model run-time (e.g. radiative heating).

During an eruption crisis, PUFF predictions have been used to estimate ash cloud movement critical to the assessment of potential impacts – for example, on aircraft flight paths. As other more complicated (and thus slower) VATD model results become available they can be incorporated into an impact analysis by volcanologists and atmospheric scientists. Thus, uncertainty analysis of VATD models in general, and of the PUFF model in particular, is important due to potential ramifications in mitigating natural hazards.

To start a simulation, PUFF requires as inputs the eruption start time and duration, the initial plume height, the vertical distribution of particles of varying size, a representative wind field, and the simulation end time (see Sec.1.2). At first, some of these parameters must be assumed, based on past activity of the volcano, or by using the Eruption Source Parameters (ESP) of Mastin et al. [8]. In a post-event assessment, more information about the eruption is known and can be incorporated into a "post-diction" or hindcast analysis. It is thought that the parameters that most significantly affect simulation results are the initial wind field, the plume height, and the particle distribution [7]; one part of the research plan proposed here is a sensitivity analysis of PUFF model inputs.

PUFF can be run using one of several numerical weather prediction (NWP) windfields [9, 10, 11, 12]. These NWP models are available at differing levels of spatial and temporal resolution; local use of the WRF (Weather Research Forecast) model will allow for customizing the model domains used for verification and validation.

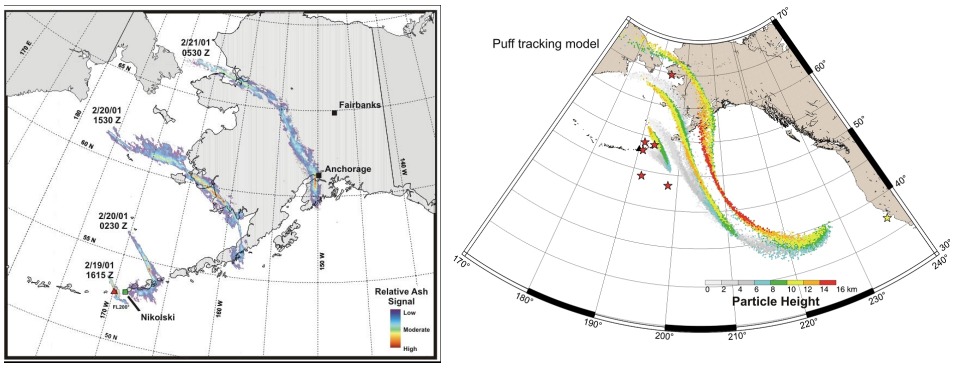
PUFF tracks a finite number of Lagrangian point particles of different sizes, whose location $R$ is propagated from timestep $k$ to timestep $k+1$ via an advection/diffusion equation

$$R_i(t_{k+1}) = R_i(t_k) + W(t_k)\Delta t + Z(t_k)\Delta t + S_i(t_k)\Delta t \tag{1}$$

Here $R_i(t_k)$ is the position vector of the $i^{th}$ particle at time $k\Delta t$, $W(t_k)$ is the local wind velocity at the location of the $i^{th}$ particle, $Z(t_k)$ is a turbulent diffusion that is modeled as a random walk, and $S_i(t_k)$ is a source term which models the fallout of the $i^{th}$ particle due to gravity. For more detailed description see [4]; source code and documentation is available at `http://puff.images.alaska.edu/monitoring.shtml`. A variety of output displays are available, including snapshots of airborne-ash concentration (relative to the number of particles released at the start of the simulation) and particle location color-coded by height. PUFF is capable of tracking multiple ash clouds simultaneously [13, 14].

### 1.1.1 PUFF Validation

PUFF has been validated against historic volcanic eruptions such as the 1992 Crater Peak vent eruption at Mount Spurr and the 2006 eruption at Mount Augustine with reasonable success [4, 14]. Webley et al. [6] performed an analysis of the most relevant VATD model input parameters on simulation outputs, making a statistical comparison between the satellite detected ash cloud and PUFF model results for the August and September 1992 events at Mount Spurr. Dean et al. [15] present a comprehensive analysis of images from GOES (Geostationary Operational Environmental Satellite), and AVHRR (Advanced Very High Resolution Radiometer), of the 2001 eruption of Mount Cleveland, Alaska. Figure 1 illustrates this comparison. Discrepancies were noted in the PUFF model output and the GOES and AVHRR images. The PUFF model, based on a wind shear at an altitude of 6 $km$ above sea level predicted the ash plume to project northwest and southeast, with an overall drift towards the northeast. However, the GOES imagery only captured the northwest




(a) GOES image

(b) PUFF Estimate

Figure 1: PUFF Estimate vs. GOES image for the 2001 Mt. Cleveland eruption. PUFF output showed wind shears at 6 km that sent particles NW and SE. However, the GOES imagery only shows the NW extension and a NE drift. A Boeing 747 pilot noted the smell of $SO_2$ at a time and location that coincides with PUFF predictions (yellow star), supporting the model prediction.

extension and northeast drift. Dean et al. state that a Boeing 747 pilot noted the smell of sulphur dioxide at a time and location that coincided with the PUFF prediction but not with satellite images [15]. Because ash particles are moved by the windfield, at a given altitude any separation of the main body of the cloud and gases such as $SO_2$ are likely to be minimal. This discrepancy between the GOES picture and AVHRR observations and PUFF results is puzzling, and may be related to satellite sensor detection limits, or environmental conditions affecting the ash signal. In any event this finding underscores the need for a systematic study of VATD model uncertainty.

Satellite images provide only a planar view of the cloud-top or, where the cloud is translucent, cloud particles from lower elevations; plots of simulation position *vs.* time (for different altitudes) can be compared with images such as in [4, 6]. Satellite remote sensing data is temporally (at best, 15 minute updates for GOES) and spatially (at best, 1 km for AVHRR) coarse. The most useful satellite data for ash cloud detection is measured in the $10 - 12$ $\mu$m wavelength, either as single band and band ratio, or in a reverse absorption/split-window [5]. Pavolonis et al. [16] show that ash clouds can also be detected through visible remote sensing datasets, which can complement the infrared information. After many years of responding to eruptions, UAF researchers have noticed variations in the accuracy of PUFF predictions when compared to satellite image and ground observations [15]. Roughly speaking, PUFF appears reasonably accurate in predicting the movement and position of volcanic clouds in space and time up to the first 12-24 hours, which is the required time period for operational ash cloud forecasts for aviation. Significant differences between satellite observations and PUFF predictions can appear owing to atmospheric characteristics and/or boundary layer topography. There are lower limits to satellite detection of volcanic ash based on concentrations and spatial resolution, as well as other environmental conditions that affect signal strength, so validation of dispersion models under these conditions is limited and the accuracy can not be evaluated [13, 15]. These issues may be important in validation studies.

Nine world-wide VAACs have been designated and networked together by the International Civil Aviation Organization, to provide advisories regarding volcanic eruptions, especially to the aviation community. Remote sensing and VATD modeling are integral parts of the operation of the VAACs.

### 1.2 BENT-PUFF

As mentioned above, to initialize a PUFF simulation a collection of particles of different sizes must be specified as a function of altitude. This distribution is often inferred from historical eruption and plume observations, and is not well constrained; see [17, 18, 19]. It is important to remember that PUFF particles are not simple surrogates for ash concentration, but are representatives of ejecta of a given size at some initial height. As such this number is a user-selected input, and affects the simulation time and resolution of the output. In addition to particle distribution and windfield, other PUFF input parameters include the coefficients of turbulent diffusion and particle fallout, both of which are estimated.

Instead of guessing the initial particle distribution as a function of height, we employ a volcanic eruption plume model called BENT to provide initial conditions; the essential features of this coupling are described in [20]. BENT solves a cross-sectionally averaged system of equations for continuity, momentum and energy balance [19, 18, 17], as a function of the eruption vent radius and speed of the ejecta. The ambient temperature at the vent, and the temperature difference between the erupting column and the ambient atmosphere, usually do not vary much in nature, and do not significantly affect plume results [17]. BENT also requires coefficients for heat capacity and re-entrainment, which are approximately known. BENT assumes a distribution of pyroclasts of different sizes, and the model equations then predict the height distribution of the various sized clasts. BENT results suggest that the interaction between plume and wind causes enhanced entrainment of air and horizontal momentum, plume bending, and a decrease in plume rise height at constant eruption rate. Thus, wind and atmospheric stratification affect the plume rise height [21]. BENT has been tested against plume rise height data, and against dispersal data [17]; the discussion in that paper corroborates that the scaling relationships derived in [22] between energy and plume rise height are valid even for energetic volcanic plumes piercing the tropopause.

Using BENT to provide initial conditions for PUFF incorporates important plume physics into our cloud transport simulations. On the one hand, physics guides our model coupling and largely determines for us how outputs from BENT feed into PUFF. On the other hand, this coupling can be thought of as substituting one set of uncertain parameters (vent size, velocity, clast size distribution) for an uncertain function (initial particle height distribution).

## 2 Uncertainty Characterization

Of course, any model used to represent the dynamics of a tephra plume reflects many assumptions and simplifications that permit derivation of a tractable model. The error inherent in any model is a result of model truncation, errors in model parameters, and errors in initial and boundary conditions. In addition to these uncertainties in inputs to the model system, there are stochastic features such as variations in the wind, which drive the motion of an ash cloud. Together these factors cause overall accuracy to degrade as the simulation evolves. We now discuss the uncertainty in our model, and data assimilation techniques can be used to update model predictions.

Mathematical models of physical processes are subject to uncertainty from several sources: parameters, initial and boundary conditions, forcing functions, are known only to certain precision. For example, BENT input parameters include vent radius, vent velocity, mean grain size and grain size variance. All of these parameters are known only approximately and only to a fixed fidelity. Furthermore, these models are but approximate representations of the underlying physical processes. For example, a more comprehensive ash cloud model might account for precipitation and particle entrainment, specifying some of the physical processes that are only approximated in the PUFF model. Such a model would require additional parameters and inputs that themselves are

not (usually) well characterized. Not only can uncertainty enter into a complex model through a number of avenues, the subsequent propagation of uncertainty through the model is also a complex process. To account for at least many of the parametric uncertainties in a model one might evaluate the governing equations with many different inputs, and average the outputs appropriately. Unfortunately, for a more comprehensive model of ash cloud motion the computational cost makes this approach infeasible.

*How are simulation outcomes affected by imprecise knowledge of the model and its inputs? Given these uncertainties, how do we forecast the evolution of a dynamical process together with its attendant uncertainty, in a computationally tractable manner?*

Propagating uncertain model inputs results in forecasts with uncertainties that can grow in time. Assimilating available observational data to refine the model forecast reduces these uncertainties. However limited sensor range and sensor inaccuracies can lead to imprecise measurements. An improved solution should be a weighted combination of simulation forecast and observation data.

*How does one design computationally tractable data assimilation tools that incorporate information from various sources while simultaneously compensating for simulation errors and observational inaccuracies?*

Despite the potential risk to property and life from inaccurate prediction of ash clouds, *there has never been a thorough quantitative assessment of toxic cloud predictions due to parametric and stochastic uncertainties.* While a detailed sensitivity analysis can relate the variations in input parameters to toxic cloud, *uncertainty analysis casts a much broader net in terms of assessing confidence of predictions based on all available information.*

A mechanism to represent the uncertainty is necessary before the model data and the sensed data can be integrated in an efficient and consistent manner. Uncertainty can be represented in different ways, including: *i) worst-case scenarios attempting to provide bounds using interval analysis[23, 24], ii) methods based on fuzzy set theory, linguistically often identified as being concerned with possibility[25], iii) evidence theory, which tries to create upper and lower bounds on the likelihood of events[26], and, iv) probabilistic or stochastic models, which offer mathematically the richest structure[27, 28, 29, 30, 31, 32].* From this short and incomplete description, one may ascertain that there is no common agreement on which of these approaches may be best, even just in a certain setting.

Probabilistic means of representing uncertainties has been explored extensively and provides the greatest wealth of knowledge which are exploited in this work.

## 2.1 Parametric Uncertainty Characterization

In the standard `BENT` and `PUFF` models, one tracks the position of representative particles of different sizes as they are transported by wind and turbulence, and the position of each particle is assumed to be a deterministic quantity. Instead of solving for the point position of the `PUFF` particles, we consider a probability distribution for each particle location. That is, the position of a particle is assumed to be a random variable, $\mathbf{x}_k$, whose time evolution is given by a stochastic differential equation (which should be thought of as generalizing the `PUFF` advection/diffusion equation):

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{\Theta}) \tag{2}$$

In this equation, $\mathbf{\Theta}$ represents uncertain but time-invariant system parameters such as the vent radius, vent velocity, mean grain size and grain size variance. The total uncertainty associated with the state vector $\mathbf{x}_k = \mathbf{x}(t_k)$ is characterized by the probability distribution function (pdf) $p(t_k, \mathbf{x}_k)$. The index $k$ denotes the discrete time-step in the evolution of $p$. A key idea of this work is to

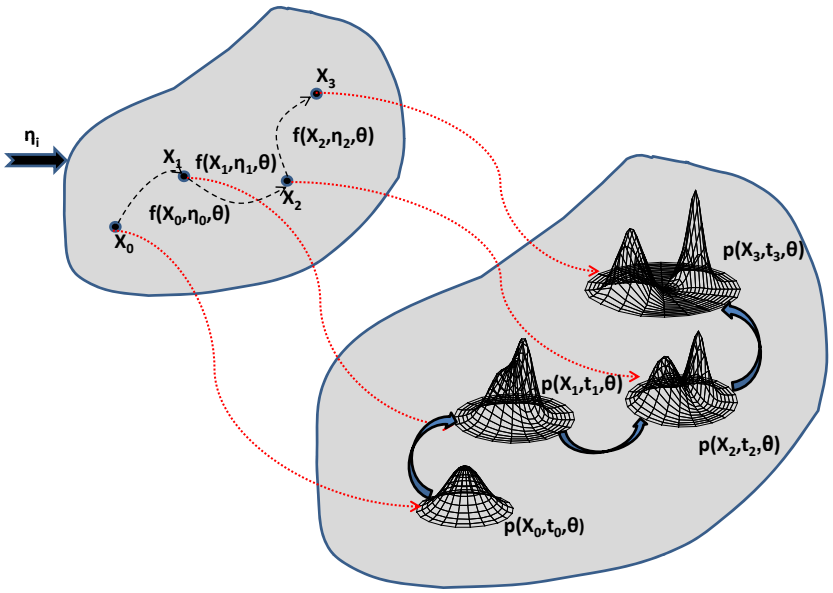



Figure 2: State and pdf transition

*replace the time evolution of state vector* $\mathbf{x}_k$ *by the time evolution of the pdf* $p(t_k, \mathbf{x}_k)$ as illustrated in Fig. 2. By computing full probability density functions, we can better monitor the space-time evolution of uncertainty, represent multi-modal distributions, incorporate complex prior models, and exploit Bayesian belief propagation.

Several approximate techniques exist in the literature to approximate the state pdf evolution [33, 34], the most popular being Monte Carlo (MC) methods [35], Gaussian closure [36], Equivalent Linearization [37], and Stochastic Averaging [38, 39]. All of these algorithms except MC methods are similar in several respects, and are suitable only for linear or moderately nonlinear systems, because the effect of higher order terms can lead to significant errors. Monte Carlo methods require extensive computational resources and effort, and become increasingly infeasible for high-dimensional dynamic systems [40].

### 2.1.1 Polynomial Chaos

The propagation of uncertainty due to time-invariant but uncertain input parameters can be approximated by a generalization of polynomial chaos (gPC), originally due to Xiu and Karniadakis[41]. gPC is an extension of the homogenous chaos idea of Wiener[42] and involves a separation of random variables from deterministic ones in the solution algorithm for a stochastic differential equation. The random variables are expanded in a polynomial expansion. These polynomials are associated with the assumed pdf for the input variables (Hermite polynomials for normally distributed parameters, Legendre for uniformly distribution, etc). Galerkin collocation is used to generate a system of deterministic differential equations for the expansion coefficients.

Let us consider a generic second order stochastic linear system:

$$\mathbf{M}(\mathbf{\Theta})\ddot{\mathbf{x}}(t,\mathbf{\Theta}) + \mathbf{C}(\mathbf{\Theta})\dot{\mathbf{x}}(t,\mathbf{\Theta}) + \mathbf{K}(\mathbf{\Theta})\mathbf{x}(t,\mathbf{\Theta}) = \mathbf{D}(\mathbf{\Theta})\mathbf{u}(t) \tag{3}$$

where $\mathbf{M} \in \mathbb{R}^{n\times n}$, $\mathbf{C} \in \mathbb{R}^{n\times n}$, $\mathbf{K} \in \mathbb{R}^{n\times n}$ and $\mathbf{D} \in \mathbb{R}^{n\times m}$. $\mathbf{\Theta} \in \mathbb{R}^r$ is a vector of uncertain system parameters which are functions of the random variable $\boldsymbol{\xi}$ with known probability distribution

function (pdf) $p(\boldsymbol{\xi})$. It is assumed that the uncertain state vector $\mathbf{x}(t, \boldsymbol{\Theta})$ and system parameters, $M_{ij}$, $C_{ij}$ and $K_{ij}$ can be written as a linear combination of basis functions, $\phi_i(\boldsymbol{\xi})$, which span the stochastic space of random variable $\boldsymbol{\xi}$.

$$x_i(t, \boldsymbol{\Theta}) = \sum_{l=0}^{N} x_{i_l}(t)\phi_l(\boldsymbol{\xi}) = \mathbf{x}_i^T(t)\boldsymbol{\Phi}(\boldsymbol{\xi}) \tag{4}$$

$$M_{ij}(\boldsymbol{\Theta}) = \sum_{l=0}^{N} m_{ij_l}\phi_l(\boldsymbol{\xi}) = \mathbf{m}_{ij}^T\boldsymbol{\Phi}(\boldsymbol{\xi}) \tag{5}$$

$$C_{ij}(\boldsymbol{\Theta}) = \sum_{l=0}^{N} c_{ij_l}\phi_l(\boldsymbol{\xi}) = \mathbf{c}_{ij}^T\boldsymbol{\Phi}(\boldsymbol{\xi}) \tag{6}$$

$$K_{ij}(\boldsymbol{\Theta}) = \sum_{l=0}^{N} k_{ij_l}\phi_l(\boldsymbol{\xi}) = \mathbf{k}_{ij}^T\boldsymbol{\Phi}(\boldsymbol{\xi}) \tag{7}$$

$$D_{ij}(\boldsymbol{\Theta}) = \sum_{l=0}^{N} d_{ij_l}\phi_l(\boldsymbol{\xi}) = \mathbf{d}_{ij}^T\boldsymbol{\Phi}(\boldsymbol{\xi}) \tag{8}$$

where $\boldsymbol{\Phi}(.) \in \mathbb{R}^N$ is a vector of polynomials basis functions orthogonal to the pdf $p(\boldsymbol{\xi})$ which can be constructed using the *Gram-Schmidt Orthogonalization Process*. The coefficients $m_{ij_l}$, $c_{ij_l}$, $k_{ij_l}$ and $d_{ij_l}$ are obtained by making use of following *normal equations*:

$$m_{ij_l} = \frac{\langle M_{ij}(\boldsymbol{\Theta}(\boldsymbol{\xi})), \phi_l(\boldsymbol{\xi})\rangle}{\langle \phi_l(\boldsymbol{\xi}), \phi_l(\boldsymbol{\xi})\rangle} \tag{9}$$

$$c_{ij_l} = \frac{\langle C_{ij}(\boldsymbol{\Theta}(\boldsymbol{\xi})), \phi_l(\boldsymbol{\xi})\rangle}{\langle \phi_l(\boldsymbol{\xi}), \phi_l(\boldsymbol{\xi})\rangle} \tag{10}$$

$$k_{ij_l} = \frac{\langle K_{ij}(\boldsymbol{\Theta}(\boldsymbol{\xi})), \phi_l(\boldsymbol{\xi})\rangle}{\langle \phi_l(\boldsymbol{\xi}), \phi_l(\boldsymbol{\xi})\rangle} \tag{11}$$

$$d_{ij_l} = \frac{\langle D_{ij}(\boldsymbol{\Theta}(\boldsymbol{\xi})), \phi_l(\boldsymbol{\xi})\rangle}{\langle \phi_l(\boldsymbol{\xi}), \phi_l(\boldsymbol{\xi})\rangle} \tag{12}$$

where $\langle u(\boldsymbol{\xi}), v(\boldsymbol{\xi})\rangle = \int_{\Omega} u(\boldsymbol{\xi})v(\boldsymbol{\xi})p(\boldsymbol{\xi})d\boldsymbol{\xi}$ represents the norm introduced by pdf $p(\boldsymbol{\xi})$ with support $\Omega$. Now, substitution of Eq. (4), Eq. (5), Eq. (7) and Eq. (8) in Eq. (3) leads to

$$\begin{aligned} e_i(\boldsymbol{\xi}) = &\sum_{j=1}^{n}\left(\sum_{l=0}^{N} m_{ij_l}\phi_l(\boldsymbol{\xi})\right)\left(\sum_{l=0}^{N}\ddot{x}_{j_l}(t)\phi_l(\boldsymbol{\xi})\right) + \sum_{j=1}^{n}\left(\sum_{l=0}^{N} c_{ij_l}\phi_l(\boldsymbol{\xi})\right)\left(\sum_{l=0}^{N}\dot{x}_{j_l}(t)\phi_l(\boldsymbol{\xi})\right) \\ &+ \sum_{j=1}^{n}\left(\sum_{l=0}^{N} k_{ij_l}\phi_l(\boldsymbol{\xi})\right)\left(\sum_{l=0}^{N} x_{j_l}(t)\phi_l(\boldsymbol{\xi})\right) - \sum_{j=1}^{m}\left(\sum_{l=0}^{N} d_{ij_l}\phi_l(\boldsymbol{\xi})\right)u_j, \quad i = 1, 2, \cdots, n \end{aligned} \tag{13}$$

Now, $n(N+1)$ time-varying unknown coefficients $x_{i_k}(t)$ can be obtained by using the Galerkin discretization process, i.e., projecting the error of Eq. (13) onto the space of basis functions $\phi_l(\boldsymbol{\xi})$.

$$\langle e_i(\boldsymbol{\xi}), \phi_l(\boldsymbol{\xi})\rangle = 0, \quad i = 0, 1, 2, \cdots, n, \quad l = 1, 2, \cdots, N \tag{14}$$

where $e_i(\boldsymbol{\xi})$ is the approximation error resulting from representing the system states by a Polynomial Chaos expansion. This leads to following set of $n(N+1)$ *deterministic differential equations:*

$$\mathcal{M}\ddot{\mathbf{x}}_p(t) + \mathcal{C}\dot{\mathbf{x}}_p(t) + \mathcal{K}\mathbf{x}_p(t) = \mathcal{D}\mathbf{u}(t) \tag{15}$$

where $\mathbf{x}_p(t) = \left\{\mathbf{x}_1^T(t), \mathbf{x}_2^T(t), \cdots, \mathbf{x}_n^T(t)\right\}^T$ is a vector of $n(N+1)$ unknown coefficients and $\mathcal{M} \in \mathbb{R}^{n(N+1)\times n(N+1)}$, $\mathcal{C} \in \mathbb{R}^{n(N+1)\times n(N+1)}$, $\mathcal{K} \in \mathbb{R}^{n(N+1)\times n(N+1)}$ and $\mathbf{D} \in \mathbb{R}^{n(N+1)\times m}$.

Let $P$ and $T_k$, for $k = 0, 1, 2, \cdots, N$, denote the inner product matrices of the orthogonal polynomials defined as follows:

$$P_{ij} = \langle \phi_i(\boldsymbol{\xi}), \phi_j(\boldsymbol{\xi}) \rangle, \quad i, j = 0, 1, 2, \cdots, N \tag{16}$$

$$T_{k_{ij}} = \langle \phi_i(\boldsymbol{\xi}), \phi_j(\boldsymbol{\xi}), \phi_k(\boldsymbol{\xi}) \rangle, \quad i, j = 0, 1, 2, \cdots, N \tag{17}$$

Then $\mathcal{M}$, $\mathcal{C}$ and $\mathcal{K}$ can be written as $n \times n$ matrix of block matrices, each block being an $(N+1) \times (N+1)$ matrix. The matrix $\mathcal{M}$ consists of blocks $\mathcal{M}_{ij} \in \mathbb{R}^{(N+1)\times(N+1)}$:

$$\mathcal{M}_{ij} = M_{ij} P, \quad i, j = 1, 2, \cdots, n \tag{18}$$

if the mass matrix is not uncertain, else, it is given by:

$$\mathcal{M}_{ij}(k, :) = \mathbf{m}_{ij}^T T_k, \quad i, j = 1, 2, \cdots, n \tag{19}$$

Similarly, for the matrices $\mathcal{C}$ and $\mathcal{K}$, the $k^{th}$ row of each of their block matrices $\mathcal{C}_{ij}, \mathcal{K}_{ij} \in \mathbb{R}^{(N+1)\times(N+1)}$ is given by,

$$\mathcal{C}_{ij}(k, :) = \mathbf{c}_{ij}^T T_k, \quad i, j = 1, 2, \cdots, n \tag{20}$$

$$\mathcal{K}_{ij}(k, :) = \mathbf{k}_{ij}^T T_k, \quad i, j = 1, 2, \cdots, n \tag{21}$$

The matrix $\mathcal{D}$ consists of blocks $\mathcal{D}_{ij} \in \mathbb{R}^{(N+1)\times 1}$:

$$\mathcal{D}_{ij} = P\mathbf{d}_{ij} \quad i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m \tag{22}$$

Eq. (4) along with Eq. (15) define the uncertain state vector $\mathbf{x}(t, \boldsymbol{\xi})$ as a function of random variable $\boldsymbol{\xi}$ and can be used to compute any order moment or cumulant of a function of uncertain state variable.

### 2.1.2 Simple Example

The method is best explained by illustrating its application to a simple model problem. Let us consider the first order system:

$$\dot{x} + kx = 0 \tag{23}$$

where $k$ is an uncertain parameter of the system which is known to lie between the interval $[a\ b]$. We assume it to be a function of random variable $\xi$ with known probability density function $p(\xi)$. Thus, the uncertain parameter $k$ can be represented as:

$$k(\xi) = \sum_{i=0}^{N} k_i \phi_i(\xi). \tag{24}$$

Furthermore, if $\xi \in [-1\ 1]$, only two terms are necessary to represent $k(\xi)$,

$$k(\xi) = k_0 + k_1 \xi, \ k_0 = \frac{a+b}{2}, \ k_1 = \frac{b-a}{2}. \tag{25}$$

This does not preclude Normal distributions, since $k_0$ and $k_1$ can represent the mean and variance of $k(\xi)$ and $\xi \in [-\infty\ \infty]$.

Now, the displacement $x$ is represented as:

$$x = \sum_{i=0}^{N} x_i(t)\phi_i(\xi) \tag{26}$$

where $\phi_i(\xi)$ represents the orthogonal polynomial set with respect to pdf $p(\xi)$, i.e.

$$\langle \phi_i(\xi), \phi_j(\xi) \rangle = \int_{\Omega} \phi_i(\xi)\phi_j(\xi)p(\xi)d\xi = c_i^2 \delta_{ij} \tag{27}$$

For example, the Legendre and Hermite polynomials constitute the orthogonal polynomial sets for uniform and normal distribution, respectively. In general, these polynomials can be constructed by making use of *Gram-Schmidt Orthogonalization process*. Now, substituting for $x$ and $k$ from Eqs. (26) and (24) in Eq. (23) leads to

$$\sum_{i=0}^{N} \phi_i(\xi)\dot{x}_i + (k_0\phi_0(\xi) + k_1\phi_1(\xi)) \sum_{i=0}^{N} \phi_i(\xi)x_i = 0 \tag{28}$$

Using the Galerkin projection method, the dynamics of $x_i$ can be determined. Making use of the fact that system equation error due to polynomial chaos approximation (Eq. (28)) should be orthogonal to basis function set $\phi_j(\xi)$, we arrive at the equation:

$$\mathcal{M} \begin{Bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_N \end{Bmatrix} + \mathcal{K} \begin{Bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{Bmatrix} = 0 \tag{29}$$

where the elements of the $\mathcal{M}$ matrix are

$$\mathcal{M}_{ij} = \langle \phi_i(\xi), \phi_j(\xi) \rangle = \int_{\Omega} \phi_i(\xi)\phi_j(\xi)p(\xi)d\xi = c_i^2 \delta_{ij} \text{ where } i, j = 0, 1, 2...N \tag{30}$$

and the elements of the $\mathcal{K}$ matrix are given by

$$\mathcal{K}_{ij} = k_0\langle \phi_i(\xi), \phi_j(\xi) \rangle + k_1\langle \xi\phi_i(\xi), \phi_j(\xi) \rangle \tag{31}$$

Now, making use of the fact that every orthogonal polynomial set satisfies a three-term recurrence relation[43]:

$$\xi\phi_n(\xi) = \frac{a_n}{a_{n+1}}\phi_{n+1}(\xi) + \frac{c_n^2}{c_{n-1}^2}\frac{a_{n-1}}{a_n}\phi_{n-1}(\xi) \tag{32}$$

where $a_n$ and $a_{n-1}$ are the leading coefficients of $\phi_n(\xi)$ and $\phi_{n-1}(\xi)$, respectively. Now, making use of this recurrence relationship, the elements of the $\mathcal{K}$ matrix are given by

$$\mathcal{K}_{ii} = k_0\langle \phi_i(\xi), \phi_j(\xi) \rangle = k_0 c_i^2 \tag{33}$$

$$\mathcal{K}_{i,i+1} = k_1\langle \phi_{i+1}(\xi), \phi_j(\xi) \rangle = k_1 c_{i+1}^2 \frac{a_i}{a_{i+1}} \tag{34}$$

$$\mathcal{K}_{i,i-1} = k_1 \frac{c_i^2}{c_{i-1}^2}\langle \phi_{i-1}(\xi), \phi_j(\xi) \rangle = k_1 c_i^2 \frac{a_{i-1}}{a_i} \tag{35}$$

Thus given the initial value at time $t_0$, we can compute the distribution for $x(t,\xi) = \sum_{i=0}^{N} x_i(t)\phi_i(\xi)$ for any time $t$ by integrating Eq. (29) to obtain the coefficients $x_i(t)$. Assuming $N$ is reasonably small (5 to 10) we have effectively replaced the solution of single equation Eq. (23) with the solution of a coupled set of 5 to 10 equations that directly yield an approximation of the probability distribution of the output. Note that complex nonlinearities in the equation will immediately transform into very complex expressions for the terms in Eq. (28). Solution of the coupled equation set can now become very difficult for the standard methods of solving such equations. Moreover existing computational tools cannot be used without extensive modification.

### 2.1.3 A Simple Extension - Polynomial Chaos Quadrature

To avoid these difficulties, Dalbey et al. have proposed a different formulation [44] known as polynomial chaos quadrature (PCQ). PCQ replaces the projection step of the PC with numerical quadrature. The resulting method can be viewed as a MC-like evaluation of system equations, but with sample points selected by quadrature rules. To illustrate the key idea, let us reconsider Eq. (28):

$$\sum_{i=0}^{N} \phi_i(\xi)\dot{x}_i + (k_0\phi_0(\xi) + k_1\phi_1(\xi)) \sum_{i=0}^{N} \phi_i(\xi)x_i = 0 \tag{36}$$

The projection step of PC yields:

$$\sum_{i=0}^{N} \langle \phi_i(\xi), \phi_j(\xi)\rangle \dot{x}_i + \sum_{i=0}^{N} k_0 x_i \langle \phi_0(\xi)\phi_i(\xi), \phi_j(\xi)\rangle + \sum_{i=0}^{N} k_1 x_i \langle \phi_1(\xi)\phi_i(\xi), \phi_j(\xi)\rangle = 0 \tag{37}$$

In the previous section, we evaluated various projection integrals analytically, however, the starting point of the PCQ methodology is to replace the exact integration with respect to $\xi$ by numerical integration. The familiar Gauss quadrature method makes a good choice for most cases. This yields:

$$\langle \phi_i(\xi), \phi_j(\xi)\rangle = \int \phi_i(\xi)\phi_j(\xi)p(\xi)d\xi \approx \sum_{q=1}^{M} w_q \phi_i(\xi_q)\phi_j(\xi_q) \tag{38}$$

$$\langle \phi_0(\xi)\phi_i(\xi), \phi_j(\xi)\rangle = \int \phi_0(\xi)\phi_i(\xi)\phi_j(\xi)p(\xi)d\xi \approx \sum_{q=1}^{M} w_q \phi_0(\xi_q)\phi_i(\xi_q)\phi_j(\xi_q) \tag{39}$$

$$\langle \phi_1(\xi)\phi_i(\xi), \phi_j(\xi)\rangle = \int \phi_1(\xi)\phi_i(\xi)\phi_j(\xi)p(\xi)d\xi \approx \sum_{q=1}^{M} w_q \phi_1(\xi_q)\phi_i(\xi_q)\phi_j(\xi_q) \tag{40}$$

$$\tag{41}$$

where $M$ is the number of quadrature points used. Substitution of aforementioned approximation of stochastic integral in Eq. (37) and interchanging summation and differentiation leads to

$$\frac{d}{dt}\left(\sum_{q=1}^{M}\underbrace{\sum_{i=1}^{N}x_i\phi_i(\xi_q)}_{x(t,\xi_q)}\phi_j(\xi_q)w_q\right)+\sum_{q=1}^{M}\sum_{i=0}^{N}\underbrace{(k_0\phi_0(\xi_q)+k_1\phi_1(\xi_q))}_{k(\xi_q)}\underbrace{x_i\phi_i(\xi_q)}_{x(t,\xi_q)}\phi_j(\xi_q)w_q=0 \tag{42}$$

$$\frac{d}{dt}\left(\sum_{q=1}^{M}x(t,\xi_q)\phi_j(\xi_q)w_q\right)+\sum_{q=1}^{M}k(\xi_q)x(t,\xi_q)\phi_j(\xi_q)w_q=0 \tag{43}$$

Integrating with respect to time $t$ yields:

$$\sum_{q=1}^{M}\left(x(t,\xi_q)-x(t_0,\xi_q)\right)\phi_j(\xi_q)w_q+\int_{t_0}^{t}\sum_{q=1}^{M}k(\xi_q)x(t,\xi_q)\phi_j(\xi_q)w_q dt=0 \tag{44}$$

Interchanging the order of time integration and quadrature summation leads to

$$\sum_{q=1}^{M}\left[x(t,\xi_q)-\underbrace{x(t_0,\xi_q)+\int_{t_0}^{t}k(\xi_q)x(t,\xi_q)dt}_{\mathcal{X}(t_0,t,\xi_q)}\right]\phi_j(\xi_q)w_q=0 \tag{45}$$

Note that the time integral expression in the aforementioned equation can be evaluated by a simple deterministic evaluation of the model equation with a specific instance of the random variable $\xi_q$. Thus the process of evaluating the statistics on the output of the system reduces to sampling the chosen input points guided by quadrature method. The coefficients of the PC expansion can be obtained as:

$$\sum_{q=1}^{M}x(t,\xi_q)\phi_j(\xi_q)w_q=\sum_{q=1}^{M}\mathcal{X}(t_0,t,\xi_q)\phi_j(\xi_q)w_q \tag{46}$$

Now, making use of the orthogonality property of the basis function, this leads to:

$$x_i(t)=\frac{1}{c_i^2}\sum_{q=1}^{M}\mathcal{X}(t_0,t,\xi_q)\phi_j(\xi_q)w_q,\quad c_i^2=\int_{\Omega}\phi_i(\xi)\phi_i(\xi)p(\xi)d\xi \text{ where } i,j=0,1,2...N \tag{47}$$

Hence, the resulting method can be viewed as a MC-like evaluation of system equations, but with sample points selected by quadrature rules. More generally, PCQ approximates the moment of system state $\dot{\mathbf{x}}=f(t,\mathbf{x},\boldsymbol{\Theta})$ as:

$$\langle\mathbf{x}(t)^n\rangle=\int_{\Omega}\left(\int_0^t\dot{\mathbf{x}}dt\right)^n p(\boldsymbol{\xi})d\boldsymbol{\xi}=\int_{\Omega}\left(\int_0^t f(t,\mathbf{x},\boldsymbol{\Theta})dt\right)^n p(\boldsymbol{\xi})d\boldsymbol{\xi} \tag{48}$$

For a fixed value of parameter $\boldsymbol{\Theta}=\boldsymbol{\Theta}_q$, the time integration can be performed using deterministic run of `Bent` and `PUFF`. Integration (by PCQ) over the uncertain inputs determines the state pdf. Expressed mathematically,

$$\langle \mathbf{x}(t)^n \rangle = \sum_q w_q \left( \int_0^t f(t, \mathbf{x}, \mathbf{\Theta}_q) dt \right)^n \tag{49}$$

In another words, the output moments can be approximated as a weighted sum of the output of simulations run at carefully selected values of the uncertain input parameters (namely the quadrature points). The natural choice for these quadrature points is the Gaussian quadrature points which are defined by choosing the points optimally in the sense of maximizing the degree of polynomial function that integrates exactly. The classic method of Gaussian quadrature exactly integrates polynomials up to degree $2P+1$ with $P+1$ quadrature points. The tensor product of 1-dimension quadrature points is used to generate quadrature points in general multi-dimension parameter space. As a consequence of this, the number of quadrature points increases exponentially as number of input parameter increases. It should be noted that this PCQ approach can still suffer from under-integration error if an insufficient number of samples are used. This necessitates the need for an adaptive or nested quadrature scheme to successively refines the accuracy by increasing the number of sample points such as Clenshaw-Curtis quadrature method[45, 46] for numerical integration.

## 2.2 Results and Discussion

To illustrate the effectiveness of the PCQ method, we consider the problem of volcanic ash dispersion for Eyjafjallajökull volcano eruption in April 2010. The `BENT` integral eruption column model was used to produce eruption column parameters (mass loading, column height, grain size distribution) given a specific atmospheric sounding and source conditions [19]. `BENT` takes into consideration atmospheric (wind) conditions as given by atmospheric sounding data. Thus plume rise height is given as a function of volcanic source and environmental conditions. The `PUFF` Lagrangian VATD model was used to propagate ash parcels in a given wind field (NCEP Reanalysis)[4]. `PUFF` takes into account dry deposition as well as dispersion and advection. Polynomial chaos quadrature (PCQ) was used to select sample points and weights in the uncertain input space of vent radius, vent velocity, mean particle size and particle size variance. Table 1 lists the selected distribution functions for the input parameter space. The first step is to produce the outputs and their weights for `PUFF` from `BENT` given a 4-vector of uncertain `BENT` input parameters of vent radius, vent velocity, mean grain size, and grain size variance. Each `BENT` output is then propagated through `PUFF` for a period of five days.

It is important to note that near-vent features are not captured in detail by `BENT`, and in fact, need not be. Near-vent observations indicate that eruptive pulses were characterized by ejection of an initial gas rich cap, followed by more densely laden steady flow.

Model output was compared with Meteosat-9 SEVIRI retrievals of plume height. Mean and standard deviation of ash top-height were obtained from ensemble PCQ runs and the computed footprints are compared to SEVIRI data by using the following three metrics:

$$\text{Dice} = \frac{\text{Area of Intersection of Satellite Image and PCQ Estimate}}{\text{Area of Union of Satellite Image and PCQ Estimate}} \tag{50}$$

$$\text{PCQ given Sat} = \frac{\text{Area of Intersection of Satellite Image and PCQ Estimate}}{\text{Area of Satellite Image}} \tag{51}$$

$$\text{Sat given PCQ} = \frac{\text{Area of Intersection of Satellite Image and PCQ Estimate}}{\text{Area of PCQ Estimate}} \tag{52}$$

Table 2 shows that PCQ, using a measure of mean plus three standard deviations in three-dimensional ash particle location, predicts location with greater than 98% probability ("PCQ given

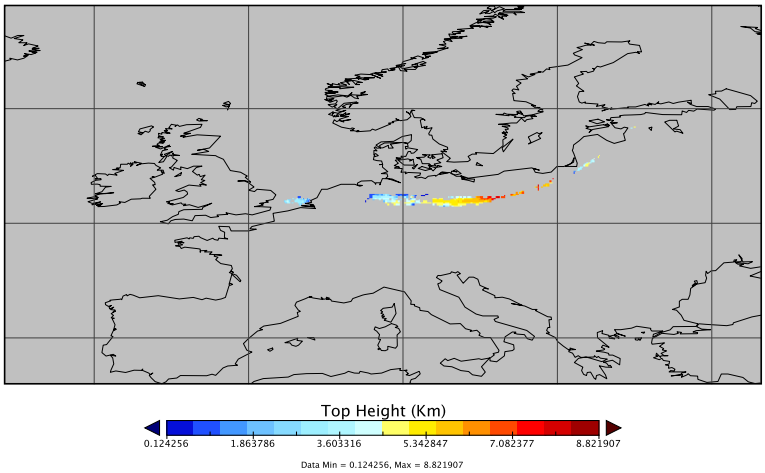
Table 1: Eruption source parameters based on observations of Eyjafjallajökull volcano and information from other similar eruptions of the past.

| Parameter | Value range | PDF | Comment |
|---|---|---|---|
| Vent radius, $b_0$, m | 65-150 | Uniform, + definite | Measured from radar image of summit vents on 14 April 2010, from 'Eruption in Eyjafjallajökull' - `http://www2.norvol.hi.is /page /ies _Eyjafjallajokull _eruption` |
| Vent velocity, $w_0$, m/s | Range: 45-124 | Uniform, + definite | M. Ripepe, "Ash Dispersal Forecasting and Civil Aviation," Geneva, Switzerland, 2010, presentation |
| Mean grain size, $Md_\varphi$ | 2 boxcars: 1.5-2 and 3-5 | Uniform, $\in \mathbb{R}$ | [47], Table 1, vulcanian and phreatoplinian. A. Hoskuldsson, Eyjafjallajökull Eruption Workshop, 09/2010, presentation, quote: 'vulcanian with unusual production of fine ash'. |
| $\sigma_\varphi$ | $1.9 \pm 0.6$ | Uniform, $\in \mathbb{R}$ | [47], Table 1, vulcanian and phreatoplinian |

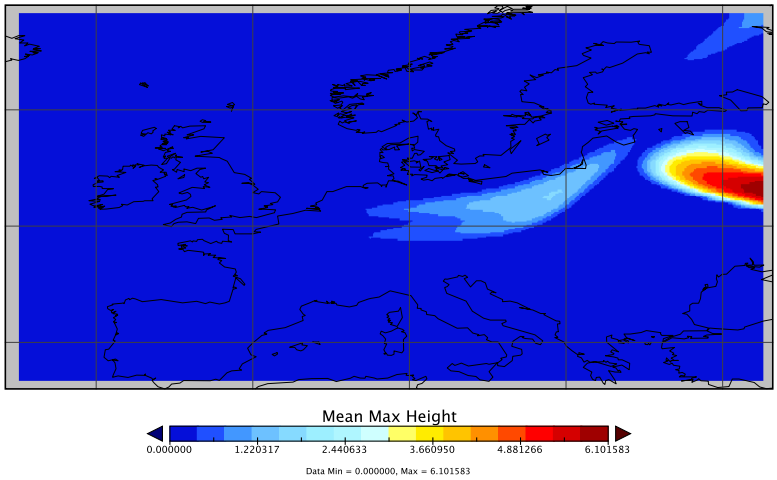
Table 2: Comparison of footprint forecast for April 16 (mean + 3× statndard deviation) and satellite data using $9^4$ and $13^4$ points of Clenshaw Curtis quadrature rule based PCQ.

| Metric of Footprint | 0<height<5km $9^4$ points | 0<height<5km $13^4$ points | 5km<height<10km $9^4$ points | 5km<height<10km $13^4$ points |
|---|---|---|---|---|
| Dice | 0.7367 | 0.7389 | 0.9469 | 0.9479 |
| PCQ given Sat | 0.9870 | 0.9873 | 0.9905 | 0.9905 |
| Sat given PCQ | 0.7714 | 0.7460 | 0.9556 | 0.9565 |

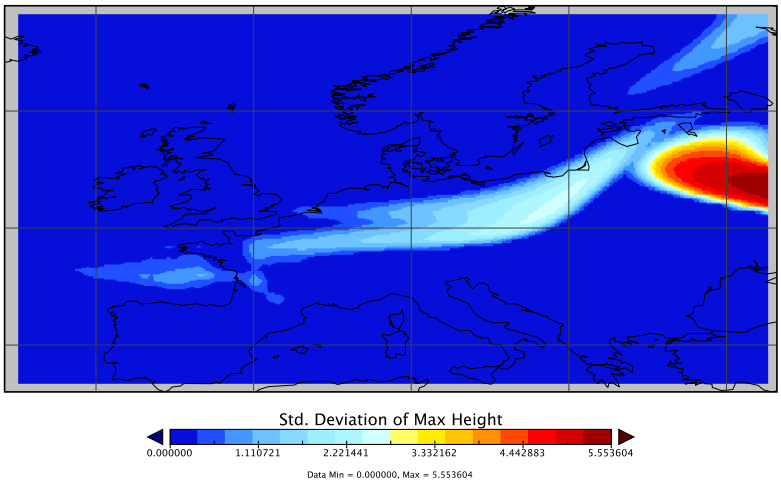
Sat" row in the table). The number of "false positives" is not unduly high, yielding no incorrect estimations of ash presence 77% of the time at low altitudes, and above 95% of the time at altitudes greater than 5km ("Sat given PCQ" row in table). Numerical methodology of the type proposed here must always be tested for consistency i.e. independence from discretization parameters, sample sizes etc. For this set of simulations we have compared the simulation outputs for $9^4$ and $13^4$ samples (quadrature points) and the results indicate such consistency (See Table 2). The comparison of using $10^5$, $4 \times 10^6$ and $10^7$ particles in the `PUFF` simulation also indicated that the choice of $4 \times 10^6$ was adequate. Furthermore, Fig. 3 shows the top height footprint obtained from SEVIRI data and mean and standard deviation of ash top height using PCQ ensemble runs. From these figures, it is clear that PCQ ensembles yield a valid estimate of mean and standard deviation of ash footprints consistent with metrics listed in Table 2. The question arises however, whether even PCQ runs can provide probabilistic concentrations useful in VAAs? These results suggest that, so long as the model range of the mass loading at least encompasses the true value, it may be possible to estimate downwind mass loading within an order of magnitude with very little knowledge of actual source conditions. As more information about source conditions is used, the uncertainty can be decreased.




(a) Top Height (SEVIRI Data)




(b) Mean Top Height (PCQ Ensemble Runs)




(c) Standard Deviation of Top Height (PCQ Ensemble Runs)

Figure 3: Meteosat-9 SEVIRI data products compared with model output for a $9^4$ run implementation of PCQ ClenshawCurtis sampling of the input space.

# 3 Data Assimilation for Reducing Uncertainty

Of course using any sensor data that might become available to correct and refine the dynamical model forecast will reduce the uncertainty of predictions. Given a prediction of the state variable $\mathbf{x}_k$, standard Bayesian algorithms assume a measurement model $\mathbf{h}$ to obtain the measurement $\mathbf{y}_k$:

$$\mathbf{y}_k = \mathbf{h}(t_k, \mathbf{x}_k) + \mathbf{v}_k, \ \mathbf{x}_k = \mathbf{x}(t_k) \tag{53}$$

where, the nonlinear function $\mathbf{h}(.)$ captures the sensor model and $\mathbf{v}_k$ is the measurement noise with prescribed likelihood function $p(\mathbf{y}_k|\mathbf{x}_k)$. This model-data fusion process is well documented in meteorology applications where a variety of data assimilation methods are used operationally to improve the quality of the Numerical Weather Prediction (NWP) forecast[48, 49, 50, 51, 52].

Standard Bayesian tools will be employed to incorporate observational data. Using the dynamic state evolution sketched above as a forecasting tool PCQ, the state pdf can be updated using the Bayes' rule on the arrival of a measurement data:

$$p(\mathbf{\Theta}|\mathbf{Y}_k) = \frac{p(\mathbf{y}_k|\mathbf{\Theta})p(\mathbf{\Theta}|\mathbf{Y}_{k-1})}{\int p(\mathbf{y}_k|\mathbf{\Theta})p(\mathbf{\Theta}|\mathbf{Y}_{k-1})\mathrm{d}\mathbf{\Theta}} \tag{54}$$

Here, $p(\mathbf{\Theta}|\mathbf{Y}_{k-1})$ represents the prior pdf (the computed distribution from PCQ), $p(\mathbf{y}_k|\mathbf{\Theta})$ is the likelihood that we observe $\mathbf{y}_k$ given the parameter $\mathbf{\Theta}$ and $p(\mathbf{\Theta}|\mathbf{Y}_k)$ represents the posterior pdf of $\mathbf{\Theta}$. The quadrature weights are updated by computing the posterior mean:

$$\int \mathbf{\Theta} p(\mathbf{\Theta}|\mathbf{Y}_k) d\mathbf{\Theta} \approx \sum_q w_q^+ \mathbf{\Theta}(t_k, \xi_q) = \frac{1}{c} \int \mathbf{\Theta} p(\mathbf{y}_k|\mathbf{\Theta}) p(\mathbf{\Theta}|\mathbf{Y}_{k-1}) \approx \frac{1}{c} \sum_q w_q^- \mathbf{\Theta}(t_k, \xi_q) p(\mathbf{y}_k|\mathbf{\Theta}(t_k, \xi_q) \tag{55}$$

where $w_q^+$ represents the unknown posterior weights for quadrature points while $w_q^-$ represents the known priori weights. Now equating the coefficients for $\mathbf{\Theta}(t_k, \xi_q)$ leads to

$$w_q^+ = \frac{w_q^- \beta_q}{\sum_{i=1}^{N} w_q^- \beta_q}, \quad \beta_q = p(\mathbf{y}_k|\mathbf{\Theta}(t_k, \xi_q)) \tag{56}$$

## 3.1 Results and Discussion

To illustrate the key idea of data assimilation for volcanic ash dispersion, we consider mean particle size to be a random variable while considering vent radius, vent velocity and particle size sigma to be deterministic quantities given as:

| | |
|---|---|
| Vent Radius = 91.236m | Vent Velocity = 45m/sec. |
| Mean Particle Size = $\mathcal{U}(1.5, 5)$ | Particle Size Sigma = 5.848 |

According to the PCQ formulation, we expand mean particle size in terms of the uniform random variable $\xi \in \mathcal{U}(-1, 1)$, i.e., $\mathbf{\Theta} = 3.25 + 1.75\xi$. Following runs of `BENT` at the quadrature points, each `BENT` output is then propagated through `PUFF`, which was then run for a real-time period of five days. The outputs from `PUFF` were then combined to produce the ensemble by applying the appropriate weight to each deterministic `BENT` and `PUFF` run.

For simulation sake, we create synthetic plume top-height footprints depicting artificial satellite imageries for $\mathbf{\Theta} = 2.3672$ which corresponds to one of the quadrature point (Case 1). The footprints

obtained by ensemble runs are compared to artificial satellite imagery using the dice metric. Fig. 4(a) shows the plot of the dice metric as a function of quadrature points. For computation sake, we assume the likelihood function to be gaussian with mean zero and standard deviation being 0.1 units. Using Eq. (56), we update the posterior weights using the synthetic satellite imagery. As expected, the posterior weight for quadrature point corresponding to the true parameter value shoots close to one as shown in Fig. 4(b)

To further validate the methodology, we create another top-height footprint for a random sample of $\xi$ yielding true value of $\mathbf{\Theta}$ to be 3.7742 (Case 2). Fig. 5(a) shows the plot of the dice metric as a function of quadrature points while Fig. 5(b) shows the plots for prior and posterior quadrature weights. As expected, the posterior weight for quadrature points near to the true parameter value increases. The multiple peaks in Fig. 5(b) corresponds to non-unique footprint shape due to variation in mean grain size as depicted in Fig. 5(a).

## References


[1] S.N. Carey and R.S.J. Sparks. Quantitative models of the fallout and dispersal of tephra from volcanic eruption columns. *Bull. Volcanology*, 48:109–125, 1986.

[2] T. Suzuki. *A theoretical model for dispersion of tephra*, pages 95–116. Terra Scientific Publishing, Tokyo, 2005.

[3] H.L. Tanaka. Development of a prediction scheme for the volcanic ash fall from redoubt volcano. In *First Int'l Symposium on Volcanic Ash and Aviation Safety*, page 58, Seattle, 1991.

[4] C. Searcy, K.G. Dean, and B. Stringer. PUFF: A volcanic ash tracking and prediction model. *J. Volcanology and Geophysical Research*, 80:1–16, 1998.

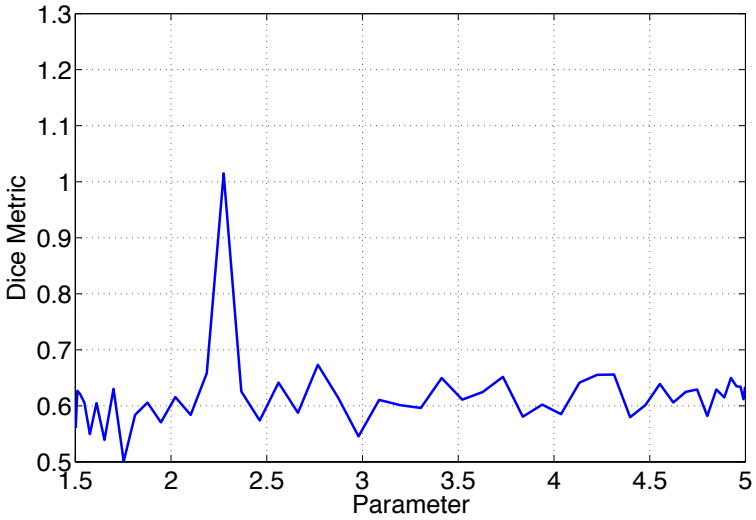
[5] A.J. Prata. Observations of volcanic ash clouds in the $10 - 12$ $m$. window using AVHRR/2 data. *International J. of Remote Sensing*, 10, 1989.

[6] P.W. Webley, B.J.B. Stunder, and K. Dean. Significant eruption source parameter(s) for operational ash cloud transport and dispersion models. *J. of Volcanology and Geothermal Research*, 186:108–119, 2009. Special Issue on Volcanic Ash Clouds, L. Mastin and P.W. Webley (eds.).

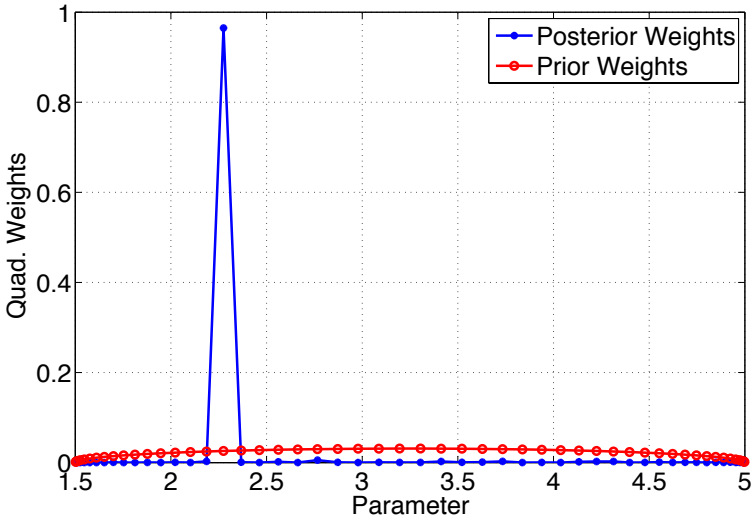
[7] R. Peterson, P.W. Webley, R. DAmours, R. Servranckx, R. Stunder, and K. Papp. *Volcanic Ash Could Dispersion Models.*

[8] L.G. Mastin, M. Guffanti, R. Servanckx, P.W. Webley, S. Barostti, K. Dean, R. Denlinger, A. Durant, J.W. Ewert, C.A. Gardner, A.C. Holliday, A. Neri, W.I. Rose, D. Schneider, L. Siebert, B. Stunder, G. Swanson, A. Tupper, A. Volentik, and A.F. Waythomas. A multidisciplinary effort to assign realistic source parameters to models of volcanic ash-cloud transport and dispersion during eruptions. *J. of Volcanology and Geothermal Research*, 186:10–21, 2009. Special issue on Volcanic Ash Clouds; L. Mastin and P.W. Webley (eds.).

[9] National Center for Environmental Prediction (2009). Unidata online access to the operational Global Forecasting System (GFS) numerical weather prediction model. `http://motherlode.ucar.edu:8080/thredds/catalog/fmrc/NCEP/GFS/Global_0p5deg/catalog.html`.

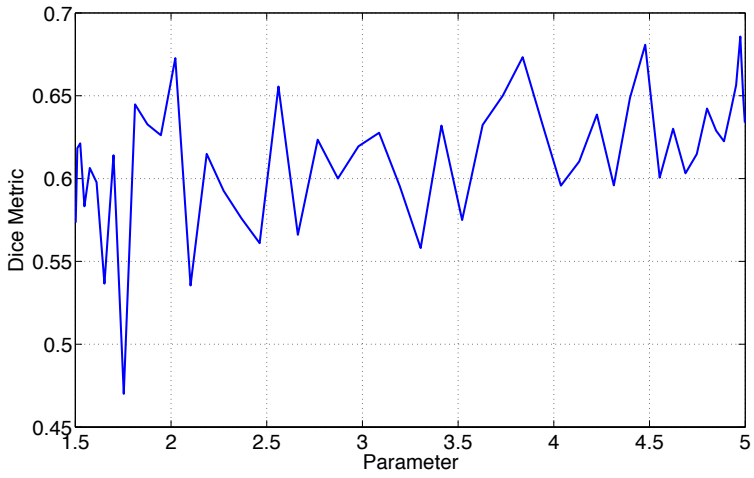



(a) Dice Metric

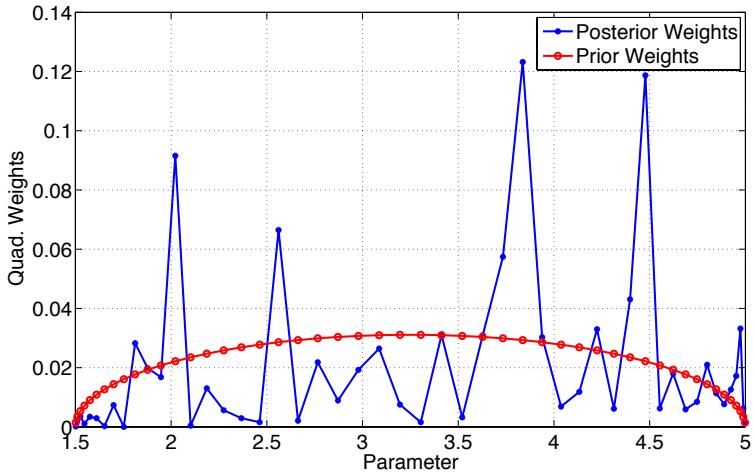



(b) Prior and Posterior Weights

Figure 4: Simulation Results for Case 1




(a) Dice Metric




(b) Prior and Posterior Weights

Figure 5: Simulation Results for Case 2

[10] National Center for Environmental Prediction (2009). Unidata online access to the operational North American Mesoscale (NAM) numerical weather prediction model. `http://motherlode.ucar.edu:8080/thredds/catalog/fmrc/NCEP/NAM/Alaska_11km/catalog.html`.

[11] United States Navy Fleet Numerical Meteorology and Oceanography Center (2009). Online access to the Navy Operational Global Atmospheric Prediction Systemnumerical weather prediction model. `https://www.fnmoc.navy.mil/public`.

[12] Weather Research and Forecasting Model (2009). Model introduction and access to real-time forecasts. `http://www.wrf-model.org`.

[13] K. Sassen, J. Zhu, P.W. Webley, K. Dean, and P. Cobb. Volcanic ash plume identification using polarization Lidar:Augustine eruption, Alaska. *Geophysical Research Letters*, 34:L08803, 2007.

[14] P.W. Webley, K. Dean, J. Dehn, J.E. Bailey, and R. Peterson. Volcanic ash dispersion modeling of the 2006 eruption of Augustine Volcano . *USGS Professional Paper: Augustine Volcano 2006 eruption*.

[15] K.G. Dean, J. Dehn, K.R. Papp, S. Smith, P. Izbekov, R. Peterson, C. Kearney, and A. Steffke. Integrated satellite observations of the 2001 eruption of Mt. Cleveland, Alaska. *J. of Volcanology and Geothermal Research*, 135, 2004.

[16] M.J Pavolonis, W.F. Feltz, A.K. Heidinger, G.M., and Gallina. A daytime complement to the reverse absorption technique for improved automated detection of volcanic ash. *J. Atmospheric and Oceanic Technology*, 23:1422–1444, 2006.

[17] R. S. J. Sparks, M. I. Bursik, S. N. Carey, J. S. Gilbert, L. S. Glaze, H. Sigurdsson, and A. W. Woods. *Volcanic Plumes*. John Wiley & Sons, London, 1997. 574p.

[18] M. Bursik, S. Kobs, A. Burns, O. Braitseva, L. Bazanova, I. Melekestsev, A. Kurbatov, and D. Pieri. Volcanic plumes and the wind: jetstream interaction examples and implications for air traffic. *J. of Volcanology and Geothermal Research*, 186:60–67, 2009.

[19] M. Bursik. Effect of wind on the rise height of volcanic plumes. *Geophys. Res. Lett.*, 18:3621–3624, 2001.

[20] M. Bursik, M. Jones, S. Carn, K. Dean, A.K. Patra, M. Pavolonis, E.B. Pitman, T. Singh, P. Singla, Peter Webley, H. Bjornssonxx, and M. Ripepexx. Polynomial chaos weighted ensemble modeling of the eyjafjallajokull plume of 1418 april 2010. *Science*. submitted.

[21] H. Graf, M. Herzog, J. Oberhuber, and C. Textor. The effect of environmental conditions on volcanic plume rise. *J. of Geophysical Research*, 104:24309, 1999.

[22] B. Morton, J. Turner, and G. Taylor. Gravitational turbulent convection from maintained and instantaneous sources. *Proceedings Royal Soc. London Ser. A*, 234:1–23, 1956.

[23] Y. Ben-Haim and I. Elishakoff. *Convex Models of Uncertainty in Applied Mechanics*. Elsevier, Amsterdam, 1990.

[24] H. G. Natke and Y. Ben-Haim. *Uncertainty: Models and Measures*. Akademie-Verlag, Berlin, 1997.

[25] I. Elishakoff. *Whys and Hows in Uncertainty Modelling Probability, Fuzziness, and Anti-Optimization*. Springer-Verlag, Berlin, 1999.

[26] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.

[27] R. J. Adler. *The Geometry of Random Fields*. John Wiley & sons, 1981.

[28] G. Augusti, A. Baratta, and F. Casciati. *Probabilistic Methods in Structural Engineering*. Chapman and Hall, 1984.

[29] G. Christakos. *Random Field Models in Earth Sciences*. Academic Press, 1992.

[30] M. Grigoriu. *Stochastic Calculus Applications in Science and Engineering*. Birkhauser Verlag, Basel, 2002.

[31] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, NY, 1984.

[32] S. Torquato. *Random Heterogeneous Materials*. Springer-Verlag, Berlin, 2002.

[33] A. Stuart A. Apte, M. Hairer and J. Voss. Sampling the posterior: An approach to non-Gaussian data simulation. *Physica D*, 230:50–64, 2007.

[34] J. Restrepo G. L. Eyink and F. J. Alexander. A statistical-mechanical approach to data assimilation for nonlinear dynamics: Evolution approximations. *J. Stat. Phys (in review)*.

[35] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte-Carlo Methods in Practice*. Springer-Verlag, 2001.

[36] R. N. Iyengar and P. K. Dash. Study of the random vibration of nonlinear systems by the Gaussian closure technique. *J. of Applied Mechanics*, 45:393–399, 1978.

[37] J. B. Roberts and P. D. Spanos. *Random Vibration and Statistical Linearization*. Wiley, 1990.

[38] T. Lefebvre, H. Bruyninckx, and J. De Schutter. Kalman filters of non-linear systems: A comparison of performance. *International journal of Control*, 77(7):639–653, 2004.

[39] T. Lefebvre, H. Bruyninckx, and J. De Schutter. Comment on a new method for the nonlinear transformations of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 47(8), 2002.

[40] F. Daum and J. Huang. Curse of dimensionality and particle filters. *Aerospace Conference, 2003. Proceedings. 2003 IEEE*, 4:1979–1993, March 8-15, 2003.

[41] D. Xiu and G.1 Karniadakis. The wiener-askey polynomial chaos for stochastic differential equations. *SIAM J. Scientific Computation*, 24:619–644, 2002.

[42] N. Wiener. The Homogeneous Chaos. *American J. of Mathematics*, 60(4):897–936, 1938.

[43] P. Singla and J. L. Junkins. *Multi-Resolution Methods for Modeling and Control of Dynamical Systems*. Applied Mathematics and Nonlinear Science. Chapman & Hall/CRC, Boca Raton, FL, July 2008.

[44] K. Dalbey, A.K. Patra, E.B. Pitman, M.I. Bursik, and M.F. Sheridan. Input uncertainty propagation methods and hazard mapping of geophysical mass flows. *J. of Geophysical Research*, 113:B05203, 2008.

[45] E. W. Cheney and D. Kincaid. *Numerical Mathematics and Computing*. Brooks/Cole, Pacific Grove, CA, 1999.

[46] C. W. Clenshaw and A. R. Curtis. A method for numerical integration on an automatic computer. *Numerische Mathematik*, 2:197–205, 1960.

[47] A. W. Woods and M. I. Bursik. Particle fallout, thermal disequilibrium and volcanic plumes. *Bulletin of Volcanology*, 53:559–570, 1991.

[48] R. Daley. Atmospheric data assimilation. *Cambridge University Press, Cambridge*.

[49] E. Kalnay. *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, Cambridge, 2003.

[50] J. L. Anderson. An ensemble adjustment Kalman filter for data assimilation. *Monthly Weather Review*, 129:2884–2903, 2001.

[51] J. L. Anderson and S. L. Anderson. A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Monthly Weather Review*, 127:2741–2758, 1999.

[52] G. Evensen. The ensemble Kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53:343–367, 2003.