Multi-level emulation of a volcanic ash transport and dispersion model to quantify sensitivity to uncertain parameters

It is advisable to refer to the publisher's version if you intend to cite from the work. See Guidance on citing.

To link to this article DOI: http://dx.doi.org/10.5194/nhess-18-41-2018

www.reading.ac.uk/centaur

CentAUR

Central Archive at the University of Reading

Reading's research outputs online

Natural Hazards
and Earth System
Sciences

Multi-level emulation of a volcanic ash transport and dispersion model to quantify sensitivity to uncertain parameters

Natalie J. Harvey[1], **Nathan Huntley**[2], **Helen F. Dacre**[1], **Michael Goldstein**[2], **David Thomson**[3], and **Helen Webster**[3]

[1]Department of Meteorology, University of Reading, Reading, RG6 6BB, UK
[2]Department of Mathematical Sciences, Durham University, Durham, DH1 3LE, UK
[3]Met Office, FitzRoy Road, Exeter, EX1 3PB, UK

Correspondence: Natalie J. Harvey (n.j.harvey@reading.ac.uk)

Abstract. Following the disruption to European airspace caused by the eruption of Eyjafjallajökull in 2010 there has been a move towards producing quantitative predictions of volcanic ash concentration using volcanic ash transport and dispersion simulators. However, there is no formal framework for determining the uncertainties of these predictions and performing many simulations using these complex models is computationally expensive. In this paper a Bayesian linear emulation approach is applied to the Numerical Atmospheric-dispersion Modelling Environment (NAME) to better understand the influence of source and internal model parameters on the simulator output. Emulation is a statistical method for predicting the output of a computer simulator at new parameter choices without actually running the simulator. A multi-level emulation approach is applied using two configurations of NAME with different numbers of model particles. Information from many evaluations of the computationally faster configuration is combined with results from relatively few evaluations of the slower, more accurate, configuration. This approach is effective when it is not possible to run the accurate simulator many times and when there is also little prior knowledge about the influence of parameters. The approach is applied to the mean ash column loading in 75 geographical regions on 14 May 2010. Through this analysis it has been found that the parameters that contribute the most to the output uncertainty are initial plume rise height, mass eruption rate, free tropospheric turbulence levels and precipitation threshold for wet deposition. This information can be used to inform future model development and observational campaigns and routine monitoring. The analysis presented here suggests the need for further observational and theoretical research into parameterisation of atmospheric turbulence. Furthermore it can also be used to inform the most important parameter perturbations for a small operational ensemble of simulations. The use of an emulator also identifies the input and internal parameters that do not contribute significantly to simulator uncertainty. Finally, the analysis highlights that the faster, less accurate, configuration of NAME can, on its own, provide useful information for the problem of predicting average column load over large areas.

1 Introduction

Volcanic ash is a significant hazard to aircraft, as well as human life, by reducing visibility and causing both temporary engine failure and permanent engine damage (Casadevall, 1994). The presence of ash disrupts air traffic and can result in large financial losses to the aviation industry. The eruption of the Icelandic volcano Eyjafjallajökull in April 2010 disrupted European airspace, the busiest airspace in the world, for 13 days, grounded over 95 000 flights (European Commission, 2016) and is estimated to have cost the airline industry EUR 3.3 billion (Mazzocchi et al., 2010).

In the event of an eruption, the decision to fly is informed by information provided by one of the nine Volcanic Ash Advisory Centres (VAACs). The VAACs issue hazard maps of predicted ash cloud extents based on forecasts from volcanic ash transport and dispersion simulators (VATDs). After the large-scale disruption caused by the 2010 Eyjafjallajökull eruption new guidelines were brought in by EURO-

CONTROL (the European Organisation for the Safety of Air Navigation) which require predictions of ash concentration values as well as ash cloud extents. However, there are large uncertainties in the VATD ash concentration forecasts. These uncertainties arise from a number of sources, including incomplete or inaccurate knowledge of the specific volcanic eruption (source uncertainty) and meteorological conditions and other sources of parameter and forcing function uncertainty, as well as particular physical processes being simplified or omitted (structural uncertainty) in any particular simulator. Currently, no systematic estimation of the resulting uncertainty is performed. This is a major limitation of the operational system and as such there is the danger of making incorrect decisions due to misjudging the accuracy of the simulator predictions. Mulder et al. (2017) showed that users of volcanic ash forecasts drew no-fly zones that were between 31 % smaller and 1182 % larger than areas of unsafe ash concentrations.

There have been many studies investigating the processes that control the long-range dispersion of volcanic ash. The majority of these studies focus on a small number of simulator inputs or parameters and change the parameters one at a time (OAT) to assess their impact on the predictions of volcanic ash transport. These studies test the difference between the simulator output from a control or baseline case and the output from the perturbed cases. This approach is appealing as it always calculates the change in the simulator away from a well-known baseline. Examples of studies that use this approach are Costa et al. (2006), Witham et al. (2007, 2012b), Webley et al. (2009), Dacre et al. (2011, 2015), Devenish et al. (2012a, b), Folch et al. (2012) and Grant et al. (2012). However, there are three main disadvantages of using OAT analysis. First, the amount of parameter space that is sampled quickly reduces as the number of parameters considered is increased (Saltelli and Annoni, 2010). Secondly, OAT tests ignore any interactions between parameters. For example, it is possible that perturbing two parameters separately in OAT tests might lead to negligible impacts, while the impact produced by perturbing them together might be much larger.

Finally, the analysis cannot contribute to a formal overall assessment of uncertainty: uncertainty in application of computer models includes many sources, including parameter uncertainty, measurement uncertainty, uncertainty about missing processes or about limitations in modelled processes, and so on. OAT testing does not allow a formal methodology for assessing parameter uncertainty in a way that can be combined with these other sources. The emulation method that is presented in this paper gives assessments of uncertainty that can be combined easily with other sources (although actually performing such an assessment and combination is beyond the scope of this paper).

Performing sensitivity tests that cover a wide range of parameters and parameter values for a complex simulator, such as a VATD simulator, is expensive in both time and money

because to perform such an analysis requires many simulator evaluations and hence very large computation time. This makes uncertainty quantification impractical as one can only afford a limited amount of simulator runs. Uncertainty and sensitivity analyses as well as calibration require a large number of runs. In this study we introduce the use of emulation to understand the sensitivity of an operational VATD simulator to source and internal simulator parameters.

An emulator is a simple statistical approximation of a complicated and (typically) computationally expensive function, such as a computer simulator, that can be evaluated almost instantly over the whole parameter space. The emulator provides and an associated uncertainty for this prediction (this can take the form of a full probability distribution or an expected value and variance). This enables the quantification of the impact of each simulator parameter on the prediction of the dispersion of volcanic ash. This approach has been used successfully in tsunami modelling (Sarri et al., 2012), simulating convective cloud (Johnson et al., 2015), aerosol modelling (Lee et al., 2011, 2012, 2013), galaxy formation (Vernon et al., 2010) and regional climate projections (Harris et al., 2010). In this study, the impact of the various simulator parameters can be assessed by their coefficients within the emulator. Since the emulator can be evaluated quickly, it can also be used to replace the simulator in any computationally intensive sensitivity analysis method of choice, though this step is not performed in this study. It is important to understand that emulators are used to model the behaviour of the simulator itself, when parameters are varied. That is, an emulator is designed to predict the output of the simulator under given conditions.

The relationship between the simulator output and real-world observations does not have to be considered in order to build an emulator; the "observations" used to build the emulator are observations of simulator output, not real-world measurements.

Emulators have several main uses in analysing computer simulators. They can be used for calibration to determine which parameters lead to simulator output that reasonably matches observed data. They can also be used for forecasting the future behaviour of the system in question. Finally, as in this paper, they can be used as a research tool to better understand the simulator, the role of the parameters and the interactions between them and to help guide future research priorities.

Building emulators becomes more difficult when relatively few simulator evaluations (the "data" that are used to fit emulators) are available. In many cases, however, there will be a faster and more approximate simulator available. This is true for Numerical Atmospheric-dispersion Modelling Environment (NAME). A large number of runs of this more approximate simulator can be used to build a reliable emulator (for this simulator), and then a relatively small number of evaluations of the more accurate simulator can be used to refine this into an emulator for the accurate simulator. This

approach, called multi-level emulation, is powerful but much less common in the literature. In this paper, the multi-level emulation method is adopted.

The aim of this paper is to demonstrate the potential of the multi-level emulation approach applied to a VATD simulator. We use NAME, developed at the UK Met Office (Jones et al., 2007). This simulator is used as the operational model at the London VAAC and can predict the location and concentration of volcanic ash following a volcanic eruption. In this study we focus on predicting the vertically integrated (or column) mass loadings in a particular geographical region which occurred following the 2010 Eyjafjallajökull eruption. The goal is to identify which parameters are the principle drivers of the uncertainty in the simulator's predictions of column loadings and to investigate how exactly these parameter values influence the output. The emulators are also designed for use in history matching, which is a method for determining which parameters give plausible matches to observations. This application of the emulators is deferred to a future article.

The paper is structured as follows. Section 2 describes the NAME simulator and the case study. Section 3 details the parameters that are varied in this study and the plausible ranges (as assessed by the simulator experts) for these parameters. Section 4 describes the choice of simulator runs used to build the emulators and the simulator outputs that are to be emulated. Section 5 gives an overview of the statistical methods used to build and test the emulators. The application of these methods to the case study is detailed in Sect. 6. It is intended that this paper can be used as a guide for using the methodology in other applications, so significant detail about building and validating emulators has been included. However, much of this is contained within the Appendix so that readers interested only in the details of the specific application can follow the account free of too much technical information.

2 Description of NAME and chosen case study

2.1 Model description

NAME is the VATD simulator used by the London VAAC. It is a Lagrangian particle dispersion model originally developed in response to the 1986 Chernobyl disaster. Particles, each representing a mass of volcanic ash, are released from a source. These particles are advected by 3-D wind fields provided by forecasts or analyses from a numerical weather prediction (NWP) model. The effect of turbulence is represented by stochastic additions to the particle trajectories based on estimated turbulence levels. NAME also includes parameterisations of sedimentation, dry deposition and wet deposition which are required to simulate the dispersion and removal of volcanic ash. The ash concentrations are calculated by summing the mass of particles in the model grid boxes and over a specified time period. It is important to note that some pro-

cesses affecting the eruption plume are not represented in NAME or not included in the NAME configurations used in this study. Missing processes include aggregation of ash particles, near source plume rise and processes driven by the eruption dynamics (e.g. Woodhouse et al., 2013). Note that the simulations presented in this paper were performed using NAME version 6.1.

To predict the transport and dispersion of ash, information about the volcanic eruption is required. These are known as eruption source parameters (ESPs) and include plume rise height, mass eruption rate (MER), vertical profile of the plume emissions, particle density and particle size distribution (PSD). ESPs are required to initialise the NAME simulations. Full details of the NAME setup used by the London VAAC can be found in Witham et al. (2012a). The simulations used in this study have a start time of 23:00 UTC on 7 May 2010. This start time has been chosen to ensure that NAME has had sufficient time to spin up before the chosen case study. The details of the other model parameters are discussed in Sect. 3. The ash column loadings are calculated by summing the mass of the ash in model grid boxes and averaging over 1 h. Here the model grid boxes are 0.375° latitude by 0.5625° longitude (approximately 40 km × 40 km).

2.2 Case study description

The case study chosen here is that of 14 May 2010. This is during the later phase of the Eyjafjallajökull eruption (14 April–23 May). Although this later phase of the eruption did not have as much impact on the aviation industry, it is very well observed using ground-based instruments (e.g. Pappalardo, 2013), aircraft measurements (e.g. Johnson et al., 2012) and satellites (e.g. Francis et al., 2012). Due to the large amount of observational data it is also the focus of several modelling studies (e.g. Grant et al., 2012; Devenish et al., 2012a; Dacre et al., 2013). Between 12 and 14 May, a low-pressure system moved across Iceland transporting ash cyclonically to the north and west of Iceland on 12 May, towards Europe on 13 May and to the west of Iceland on 14 May. This followed a period (approximately 7 days) of relatively settled weather dominated by a large area of high pressure in the North Atlantic. The synoptic situation at 00:00 UTC on 14 May is shown in Fig. 1a. Figure 1b shows a satellite infrared image taken by the AVHRR at 06:13 UTC on 14 May. There are high-level clouds ahead of the occluded front located between Ireland and England. Behind the front there is low-level stratus cloud.

3 Choice of uncertain input parameters

Six ESPs and 12 internal simulator parameters were selected to represent the main uncertainties affecting the simulation of the dispersion of the volcanic ash in the NAME simulator. A short description of each parameter is given below along

Table 1. Summary of the parameters, default values and ranges used in this study.

Key	Parameter name	Default value	Minimum value	Maximum value
x_1	H: height of plume at release (m)	Taken from Arason et al. (2011)	Arason et al. (2011) -2000 m	Arason et al. (2011) $+2000$ m
x_2	Source layer depth for the thin-layer source simulations (m)	1000 m	100 m	2000 m
x_3	MER: mass eruption rate (kg s^{-1})	As per Mastin et al. (2009)	Mastin et al. (2009) /3	Mastin et al. (2009) $\times$ 3
x_4	Shape parameter for the Gamma distribution for particle sizes	n/a	3	10
x_5	Scale parameter for the Gamma distribution for particle sizes (μm)	n/a	1	10
x_6	Density of the ash (kg m^{-3})	2300	1350	2700
x_7	σ_u: standard deviation of horizontal velocity for free tropospheric turbulence (m s^{-1})	0.25	0.0025	2.5
x_8	σ_w: standard deviation of vertical velocity for free tropospheric turbulence (m s^{-1})	0.1	0.001	1
x_9	τ_u: horizontal Lagrangian timescale for free tropospheric turbulence (s)	300	100	900
x_{10}	τ_w: vertical Lagrangian timescale for free tropospheric turbulence (s)	100	20	300
x_{11}	σ_m: standard deviation of horizontal velocity for unresolved mesoscale motions (m s^{-1})	0.8	0.27	1.74
x_{12}	ppt_crit: precipitation rate required for wet deposition to occur (mm h^{-1})	0.03	0	0.1
x_{13}	Scavenging coefficient parameter A for rain (s^{-1})	Below cloud: 8.4×10^{-5} In cloud: 3.36×10^{-4}	0.000001	0.01
x_{14}	Scavenging coefficient parameter A for snow (s^{-1})	Below cloud: 8.0×10^{-5} In cloud: 5.2×10^{-5}	0.000001	0.1
x_{15}	Scavenging coefficient parameter $\mathcal{B}$ for rain	0.790	0.4	1.1
x_{16}	Scavenging coefficient parameter $\mathcal{B}$ for snow	Below cloud: 0.305 In cloud: 0.790	0.2	1.2
x_{17}	R_a: dry deposition aerodynamic resistance perturbation factor	1	0.5	2
x_{18}	R_b: dry deposition Laminar sublayer resistance numerator	300	0	300

n/a is not applicable.
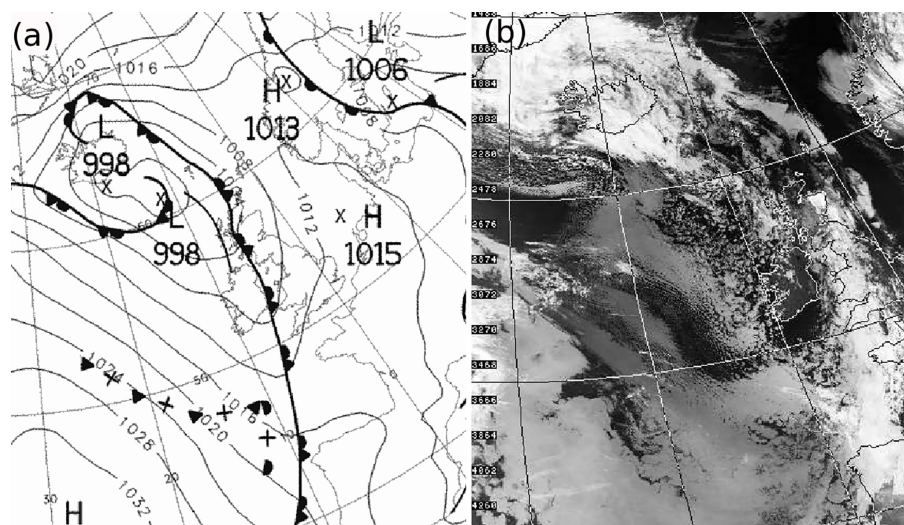
Figure 1. (a) UK Met Office surface analysis chart at 00:00 UTC on 14 May 2010. Mean sea level pressure isobars overlaid with surface fronts. **(b)** AVHRR infrared satellite image at 06:13 UTC on the 14 May 2010 provided by the Dundee satellite receiving station.

with an associated plausible range. The range represents our assessment of uncertainty on the value of each parameter. It is within these ranges that the training runs of the simulator will be performed in order to build the emulators. The uncertainty assessments were found through a small expert elicitation exercise in which information from relevant literature was combined with expert knowledge of NAME and its parameterisation schemes. Table 1 summarises the parameters and their plausible ranges. In this study we do not consider the impact of the meteorological data used to drive NAME. More detailed expert judgements on the relative plausibility of parameter choices are not required to build an emulator, although if available they could be used to adjust the training design.

3.1 Eruption source parameters

This section describes in detail the parameters specific to the eruption source and how they are perturbed in the runs used to build the statistical emulator.

3.1.1 Plume height, H

Plume height governs the height at which the ash particles are emitted into the atmosphere. This can have a large impact on the horizontal and vertical structure of the ash cloud as atmospheric wind speed and direction vary with height. Therefore to simulate realistic dispersion following an eruption it is necessary to know this height as accurately as possible. During the 2010 Eyjafjallajökull eruption information about the plume height was available from the Iceland Meteorological Office's C-band radar based at Keflavík Airport. However, there are time periods when no radar data were available. This was due to a variety of factors including the plume be-

ing obscured by meteorological cloud, missing radar scans and the fact that when the plume height was below 2.5 km it could not be detected due to the orography in the local area. When no observational plume height is available the last observed value persists until a new observation is made. In this study we will be using the data from the Keflavík radar (Arason et al., 2011) as the control plume height. This control height is then perturbed by an increment in each of the simulations used to build the emulator. The maximum and minimum increment used is ± 2 km. This is in line with observational error from the radar. Note that this study mainly focusses on one type of vertical distribution for ash at the source. This is where all ash is uniformly distributed from the volcano vent to the plume height. In other studies, for example Marenco et al. (2011), Schumann et al. (2011), Grant et al. (2012), Pappalardo (2013) and Dacre et al. (2015), a so-called "top hat" or thin-layer distribution is also used. This is where all ash is emitted in a thin layer coincident with the maximum height of the plume. Emulator results for the thin-layer distribution were also obtained but were broadly similar to those for the uniformly distributed plume presented here.

3.1.2 Mass eruption rate

Currently there is no direct method of measuring how much mass is being emitted from an erupting volcano. Therefore many VAACs use an empirical relationship between the observed plume height and the eruption rate. There are number of relationships in the literature relating these two quantities (e.g. Sparks et al., 1997; Mastin et al., 2009). In this paper the following relationship, based on Mastin et al. (2009), is used:

$$\text{MER} = 140.8 H^{4.15}, \tag{1}$$

where H is the plume height above the volcano summit in kilometres and MER represents the total MER in kilograms per second (Mastin et al., 2009; Webster et al., 2012). Here H is the perturbed plume height described in Sect. 3.1.1. Due to the empirical nature of this formulation the MER also has an associated uncertainty as the data used to form the relationship is based only on a small number of volcanoes of a similar nature (Mastin et al., 2009). To account for this the MER is perturbed by a factor between 1/3 and 3.

3.1.3 Particle size distribution

In the simulations used here, only fine ash is represented with diameters ranging from 0.1 to 100 μm separated into size size bins. The NAME default PSD (shown in Table 2) is based on observations by Hobbs et al. (1991) of ash from explosive eruptions of Mount Redoubt, St. Augustine and Mount St. Helens. The mass fraction of dispersing material is divided over the model particles within each size range. Each model particle may correspond to many actual particles of a certain diameter. The exact diameter allocated to each model particle is such that the log of the diameter is uniformly distributed within each size range making up the PSD.

The PSDs used in the simulations to build the emulator were formulated as follows. Dacre et al. (2013) present several observed PSDs for the period around 14 May 2010; it was decided to choose a range for the PSDs that included all of these. These observed PSDs can all be reasonably reproduced using gamma distributions with particular shape and scale parameters. Therefore, instead of specifying a range for the frequency associated with each particle diameter bin, a range was specified for these two parameters. For any given pair within this range, the required PSDs can easily be computed. The range for these parameters was chosen such that the observed PSDs presented in Dacre et al. (2013) could all be reconstructed to reasonable accuracy by using the gamma distribution and varying the parameters within the range.

3.1.4 Particle density

By default, the London VAAC modelling procedure assumes that ash particles are spherical and have a density of $2300 \, \mathrm{kg \, m^{-3}}$ (Bonadonna and Phillips, 2003). In this study the density is perturbed in the range of 1350–$2700 \, \mathrm{kg \, m^{-3}}$. This range of perturbation to the particle density is considered to include the uncertainty attributed to the particle shape and aggregation.

3.2 Internal simulator parameters

The long-range transport of volcanic ash can be described by two sets of processes. The first set, advection and dispersion, represents the motion of the particles. The second set, loss processes, models how the ash is removed from the atmosphere. This section describes in detail the parameterisations

Table 2. The default input source PSD used in NAME by the London VAAC.

Particle diameter (μm)	Mass fraction
0.1–0.3	0.001
0.3–1.0	0.005
1.0–3.0	0.05
3.0–10.0	0.2
10.0–30.0	0.7
30.0–100.0	0.044

and associated parameters in NAME that represent the two sets of processes.

3.2.1 Advection and dispersion parameters

In NAME particles are advected in three dimensions by winds usually provided by a NWP model, with turbulent dispersion simulated by a random walk technique which represents the turbulent velocity structures in the atmosphere. Particles are advected each time step with the change in position involving contributions from the resolved wind velocity, the turbulence and the unresolved mesoscale motions.

Free tropospheric turbulence

The diffusion due to free tropospheric turbulence is specified by a diffusivity, K, which is related to the turbulent velocities and timescales of atmospheric motions. In NAME, the along-wind and cross-wind spread are assumed to be equal, and the eddy diffusivity is further assumed to take the form $K = (\sigma_u^2 \tau_u, \sigma_u^2 \tau_u, \sigma_w^2 \tau_w)$, where σ_u and σ_w are the standard deviations of the horizontal and vertical velocity fluctuations, respectively, and τ_u and τ_w are the corresponding horizontal and vertical Lagrangian timescales. While these quantities are likely to vary in space and time, NAME simply assumes fixed values. The default values and plausible ranges for these parameters (see Table 1) are based on observations of vertical and horizontal velocity variances and diffusivities above the atmospheric boundary layer and values used in other dispersion models (Schumann et al., 1995; Dürbeck and Gerz, 1995; H. Webster and D. Thomson, personal communication, 2014). The upper limits of these parameters are representing plausible extreme values of turbulence. Note that in this study the perturbation applied to the horizontal and vertical free tropospheric turbulence parameters is proportional.

Unresolved mesoscale motions

Low-frequency horizontal eddies with scales that lie between the resolved motions of the input meteorological data and the small three-dimensional turbulent motions represented in the turbulence parameterisation scheme are parameterised separately by the unresolved mesoscale motion scheme (Web-

ster et al., 2015). As in the free tropospheric turbulence scheme the parameters governing the unresolved mesoscale motions are fixed in time and space. It is assumed that the impact of the unresolved mesoscale motions is the same in both components of the horizontal motion. The default values appropriate to the global NWP data used in this study are $\sigma_m = 0.8 \, \text{m} \, \text{s}^{-1}$ and $\tau_m = 14\,400 \, \text{s}$. These default parameters are derived from the spectral characteristics of the input meteorological data (Webster and Thomson, 2005). At long range, only the diffusivity $\sigma_m^2 \tau_m$ matters and so, to simplify the experimental design, we seek to perturb this without worrying about the values of σ_m and τ_m separately. To achieve a diffusivity range of 0.05 to 2 times the default value, we kept τ_m constant at 6120 s and varied σ_m from 0.27 to 1.74 m s^{-1} as in Table 1.

3.2.2 Loss process parameters

This section describes the parameters associated with the processes that remove ash from the atmosphere. The loss processes represented in NAME are wet deposition and dry deposition (including sedimentation). Within NAME these losses are applied on a particle basis (i.e. the mass of each particle is reduced each time step).

Wet deposition

Wet deposition is the process of ash depletion by precipitation in the atmosphere. Two main processes are involved: washout, where material is "swept out" by falling precipitation, and rainout, where ash is absorbed directly into cloud droplets as they form by acting as cloud condensation nuclei. Both of these processes are parameterised in NAME using a bulk parameterisation. The removal of ash from the atmosphere by wet deposition processes is based on the depletion equation

$$\frac{\mathrm{d}C}{\mathrm{d}t} = -\Lambda C, \tag{2}$$

where C is the ash concentration, t is time and Λ is a scavenging coefficient. The scavenging coefficient, Λ, is given by

$$\Lambda = Ar^{\mathcal{B}}, \tag{3}$$

where r is the precipitation rate in mm h^{-1} and A and $\mathcal{B}$ are parameters that vary for different types of precipitation (e.g. rain or snow) and the process being represented (e.g. washout or rainout). The values for A and $\mathcal{B}$ are based on observations and detailed cloud modelling (Maryon et al., 1999). Note that a review of the literature highlighted that the range of experimental values for snow is much more uncertain than for rain. This translates into a larger range of possible values of A and $\mathcal{B}$ for snow than rain and thus the two additional parameters in Table 1, x_{14} and x_{16}, to take account of this.

In NAME the wet deposition scheme is only used if the precipitation rate is greater than a threshold value, ppt_crit. This acts as a filter to light drizzle. The reason for applying this threshold is that historically there has been an excessive light drizzle issue in the global version of the UK Met Office NWP model (Webster and Thomson, 2014). Applying this threshold ensures that there is not an artificial over prediction of wet deposition. The default value for ppt_crit is 0.03 mm h^{-1}. In this study this threshold is varied between 0 and 0.1 mm h^{-1}.

Dry deposition

Dry deposition is the process by which material is removed from the atmosphere by transport to, and subsequent uptake by, the ground in the absence of precipitation. Dry deposition in NAME is parameterised through a deposition velocity, v_d. The flux of ash to the ground, F, is proportional to the near-surface concentration of ash, C, and is given by

$$F = v_\text{d}C, \tag{4}$$

where v_d is determined using a resistance analogy.

$$v_\text{d} = \frac{1}{R_a + R_b + R_c}, \tag{5}$$

where R_a is the aerodynamic resistance, R_b is the laminar sublayer resistance and R_c is the surface resistance (taken to be zero for particulates such as ash) (Webster and Thomson, 2011). The aerodynamic resistance, R_a, is used to specify the efficiency with which the ash is transported to the ground by turbulence. It is parameterised using a flux gradient approach and similarity theory (Maryon et al., 1999). This means that the parameterisation is strongly influenced by the prevailing meteorological conditions and thus R_a is perturbed using a scaling factor between 0.5 and 2. The laminar sublayer resistance, R_b, represents the resistance to transport across the thin quasi-laminar layer adjacent to the surface. It is determined by both the meteorological situation and particle size. The parameterisation follows the work of Underwood (2011). For small particles, smaller than 1 μm,

$$R_b = \frac{300}{u_*}, \tag{6}$$

where u_* is the friction velocity, and for larger particles

$$R_b = \frac{300}{u_* \mathcal{D}^2}, \tag{7}$$

where $\mathcal{D}$ is the particle diameter in μm. In this study the numerator of Eqs. (6) and (7) is varied between 0 and 300 to represent the range of uncertainty in the value of R_b.
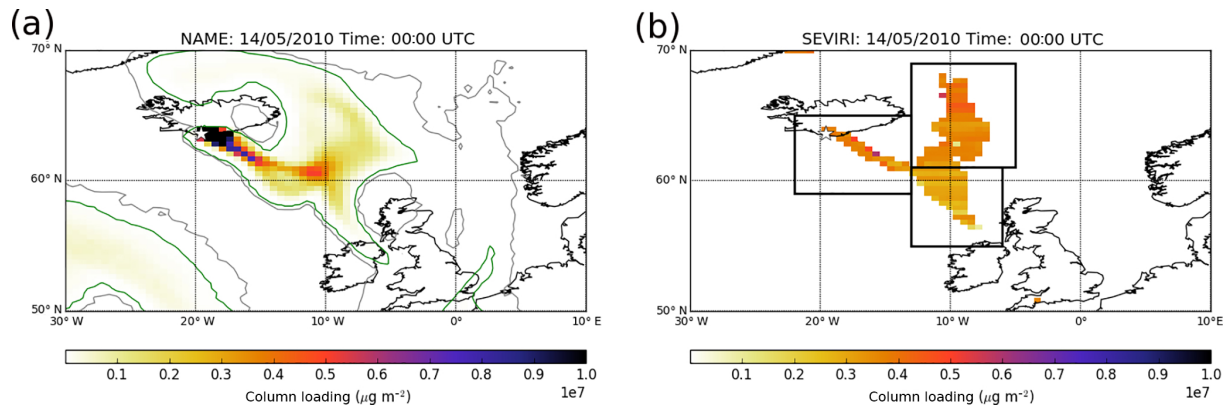
Figure 2. (a) Simulated ash column loading at 00:00 UTC on 14 May 2010 using parameters near the default values. (b) SEVIRI satellite retrieved ash column loading also at 00:00 UTC on 14 May 2010. The black boxes denote the regions over which average ash column loading is being emulated for this hour. In panel (a) column loadings of 20 000 and 2000 $\mu g\,m^{-2}$ are shown by the green and grey contours, respectively.

Sedimentation

Sedimentation of ash is represented in NAME using a sedimentation velocity, w_{sed}. This velocity is calculated using the particle diameter ($\mathcal{D}$), particle density (ρ_p) and ambient meteorological variables at the particle location (see Maryon, 1997; Webster and Thomson, 2011). In this study, w_{sed} is not perturbed as it is assumed that changes in PSD and particle density cover the range of plausible sedimentation velocities.

Distal fine ash fraction (DFAF)

The true PSD of ash particles emitted during an eruption includes extremely large particles that fall to the ground very quickly. For forecasting the effects of the eruption on aviation only the particles that will be transported large distances need to be considered. These particles form the distal ash cloud. The fraction of the total emitted ash that remains in this cloud is defined as the DFAF. DFAF is difficult to determine as it requires accurate measurements of the PSD and understanding of any aggregation processes occurring. It is also possible for DFAF to vary over time and in different parts of the ash cloud. Estimates of DFAF for the 2010 Eyjafjallajökull eruption range from 0.7 to 18.5 % (Dacre et al., 2011, 2013; Grant et al., 2012; Devenish et al., 2012b). The default DFAF assumed by the London VAAC is 5 % (Witham et al., 2012b). DFAF simply scales the modelled ash concentration and therefore does not need to be included in the analysis in this paper as the impact on the simulator output is understood perfectly.

4 Simulator runs and simulator outputs

In this study attention is focused on the ash cloud on 14 May 2010. The simulator has been set up to provide ash predictions every hour at a resolution of 0.375° latitude by 0.5625° longitude (approximately 40 km × 40 km). Figure 2a shows the simulated ash column loading at 00:00 UTC on 14 May 2010 for a choice of parameters near the default values. High column loadings are found near to and southeast of the volcano. The main plume extends towards the United Kingdom with an area of relatively low column loading in the Atlantic west of Ireland. Rather than attempt to model the entire ash cloud, it was decided to restrict attention to a small number of summaries, specifically the average ash column loading predicted over up to four large areas for each hour across a total of 24 h. The number and location of these areas changed each hour (to capture the movement of the ash cloud) with several large areas in each hour. There were a total of 75 areas analysed. These areas were chosen to cover the geographical regions where large amounts of ash were detected by satellite observations on this day. The ash column loadings retrieved using SEVIRI satellite data at 00:00 UTC on 14 May 2010 are shown in Fig. 2b. The regions used for the first hour are marked by the black boxes. The list of all regions used in the calculations can be found in Table 3.

NAME is not a fast simulator (each run of the simulator for this study took between half an hour and an hour), so it is not possible to evaluate it for very many different parameter sets. The number of NAME runs that were feasible was potentially insufficient to build the statistical models of interest.

However, a fast approximation of our standard NAME configuration could be constructed by reducing the number of particles released in the simulator from 10 000 per hour to 1000 per hour. This reduction means that "fast" simulations take between 10 and 20 min to complete. This is a significant decrease in running time but still not quick enough to apply standard global sensitivity analysis techniques such as the Morris method (e.g. Girard et al., 2014) or regional sensitivity analysis. We expect the effect of this 10-fold reduction in particle numbers to increase the particle-sampling noise in the simulations by a factor of $\sqrt{10}$. This can provide many

Table 3. Location of geographical regions used for comparison for each hour by longitude and latitude of the region corners.

	First region	Second region	Third region	Fourth region
00:00 UTC	$(-13, 61) : (-5, 69)$	$(-13, 55) : (-6, 61)$	$(-22, 59) : (-13, 65)$	
01:00 UTC	$(-14, 62) : (-6, 69)$	$(-14, 55) : (-6, 62)$	$(-22, 60) : (-14, 65)$	
02:00 UTC	$(-14, 61) : (-6, 69)$	$(-14, 54) : (-6, 61)$		
03:00 UTC	$(-14.5, 61.5) : (-6.5, 69.5)$	$(-14.5, 54) : (-4, 61.5)$		
04:00 UTC	$(-15, 62) : (-6, 70)$	$(-15, 54) : (-5, 62)$		
05:00 UTC	$(-15.5, 61) : (-6, 70)$	$(-15, 53) : (-3, 61)$		
06:00 UTC	$(-15.5, 61) : (-6, 70)$	$(-15, 53) : (-3, 61)$		
07:00 UTC	$(-17, 63.5) : (-9, 70)$	$(-14.5, 59) : (-6, 63.5)$	$(-11, 53) : (-2, 59.5)$	
08:00 UTC	$(-18, 64) : (-9, 70)$	$(-15, 61) : (-8, 64)$	$(-11, 53) : (-1, 61)$	$(-27, 63) : (-19, 66)$
09:00 UTC	$(-20.5, 64) : (-9, 71)$	$(-15, 61) : (-8, 64)$	$(-11, 53) : (-1, 61)$	$(-28, 63) : (-20, 66)$
10:00 UTC	$(-21, 64.5) : (-9, 71)$	$(-15, 61) : (-8, 64.5)$	$(-11, 53) : (-1, 61)$	$(-30, 63) : (-21, 66)$
11:00 UTC	$(-21, 63) : (-9, 71)$	$(-12, 53) : (-1, 62)$	$(-30, 63) : (-21, 66)$	
12:00 UTC	$(-22, 63.5) : (-9, 71)$	$(-12, 53) : (-1, 62)$	$(-31, 63) : (-23, 66)$	
13:00 UTC	$(-23, 63) : (-10, 71)$	$(-12, 53) : (-1, 62)$	$(-32, 63) : (-23, 66)$	
14:00 UTC	$(-24, 65) : (-17, 71)$	$(-17, 63) : (-12, 67)$	$(-12, 52) : (0, 62)$	$(-33, 62.5) : (-22, 66.5)$
15:00 UTC	$(-24, 65) : (-18, 71)$	$(-18, 63) : (-12, 67)$	$(-8, 53) : (0, 59)$	$(-33, 62.5) : (-22, 65.5)$
16:00 UTC	$(-25, 64) : (-20, 71)$	$(-20, 62) : (-12, 66)$	$(-8, 52) : (0, 58)$	$(-33, 62.5) : (-24, 66)$
17:00 UTC	$(-26, 65) : (-19, 71)$	$(-20, 62) : (-15, 65)$	$(-8, 52) : (0, 58)$	$(-34, 62.5) : (-24, 66)$
18:00 UTC	$(-28, 66) : (-19, 71)$	$(-27, 62) : (-14, 66)$	$(-7, 52) : (1, 58)$	$(-34, 62.5) : (-27, 66)$
19:00 UTC	$(-27, 62) : (-14, 67)$	$(-7, 52) : (1, 57)$	$(-34, 62.5) : (-27, 66)$	
20:00 UTC	$(-27, 62) : (-14, 67)$	$(-7, 52) : (1, 57)$	$(-36, 62.5) : (-27, 66.5)$	
21:00 UTC	$(-27.5, 61.5) : (-18, 67)$	$(-7, 52) : (1, 57)$	$(-37, 62) : (-27.5, 66.5)$	
22:00 UTC	$(-28, 63.5) : (-18, 67)$	$(-7, 51.5) : (1, 55.5)$	$(-37, 62) : (-28, 66.5)$	
23:00 UTC	$(-30, 63.5) : (-18, 66.5)$	$(-7, 51.5) : (1, 55.5)$	$(-37, 62) : (-30, 66.5)$	

approximate runs to complement the relatively few standard simulator runs. Henceforth, the fast approximation is referred to as the "fast simulator" and the standard version is referred to as the "slow simulator".

For the fast simulator runs, 1500 parameter sets were chosen using a maximin Latin hypercube design (Urban and Fricker, 2010), a method of generating multidimensional parameter sets designed to ensure good coverage of the overall parameter space. For the slow simulator runs 200 different parameter sets were chosen in the same way. Finally, the fast simulator was also run at the same 200 points as the slow simulator, so the difference between the two simulators could be assessed.

Each of the 75 regions exhibits one of three types of difference between the two simulators. In some regions, the two simulators gave almost identical results. In some regions, the two simulators gave very highly correlated results, but not identical (i.e. one simulator's output is close to simply being a multiple of the other's). In the remaining regions (typically those with relatively little ash predicted) the output of the two simulators is positively correlated, but not nearly so similar. Examples of the first and third relationships can be seen in Fig. 3. In all regions there was strong correlation between the output of the fast simulator with the output of the slow simulator, with many correlations being 0.99 and none lower than 0.7.

Before proceeding, some notation should be introduced. A particular parameter set is denoted by x, and the ith parameter within this set is x_i. Collections of parameter sets are denoted by $x_1, \ldots, x_n$. The 200 parameter sets at which the slow simulator is evaluated are denoted by $x_1, \ldots, x_{200}$. and the remaining 1500 parameter sets are denoted by $x_{201}, \ldots, x_{1700}$. The sets of parameter sets are labelled

$$\mathcal{X}_S = \{x_1, \ldots, x_{200}\}$$
$$\mathcal{X}_F = \{x_{201}, \ldots x_{1700}\}.$$

Finally, each parameter set x is normalised so that each individual parameter value lies between 0 and 1.

The slow simulator is denoted by f and the fast simulator by f'. $f(x)$ and $f'(x)$ can be seen as vectors of length 75 (the total number of geographical regions) with $f_i(x)$ being the value of the average ash column load in the ith region (for example, region 6 is the third region at 01:00 UTC on 14 May 2010; see Table 3). If $\mathcal{X}$ is a set of parameter sets, then $f(\mathcal{X})$ is the set of simulator outputs generated by applying f to each element of $\mathcal{X}$. The set of simulated outputs $f(\mathcal{X}_S)$ (that is, the set of all slow simulator outputs) is denoted by D, and $f'(\mathcal{X}_S \cup \mathcal{X}_F)$ (the set of all fast simulator output) is denoted by D'.

In this notation, the goal is then to use the evaluations D and D' to make inferences about the value of $f(x)$ for any other parameter set x and, in particular, to understand which parameters influence $f(x)$ and in what way. This will involve

(a)

(b)
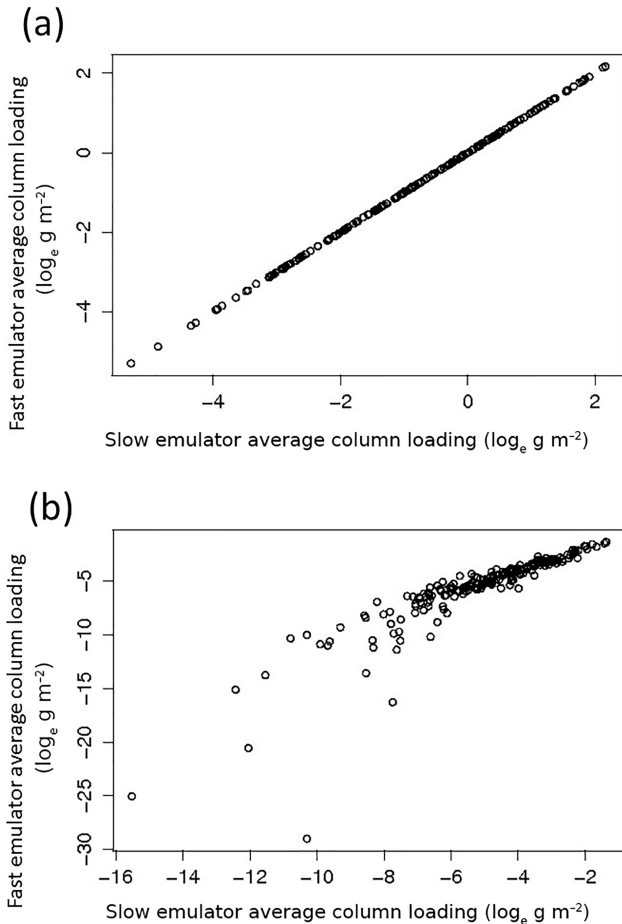
Figure 3. Relationship between the slow simulator and fast simulator output for $\mathcal{X}_\mathrm{S}$ at (first graph) the first region and (second graph) the 63rd region (third region at 19:00 UTC). The 63rd region has the lowest correlation between fast and slow simulator output.

building a statistical approximation for f, termed an "emulator". The next section describes the general form of such a model and the statistical framework used to make inferences from the simulator outputs D and D'.

5   Statistical methods

5.1   Emulation

An emulator is a simple statistical approximation of an expensive function $f(\boldsymbol{x})$ (for instance, a simulator), built using a (often small) collection of simulator runs $f(\boldsymbol{x}_i)$, which can be thought of as "data" or "observations". There are several desirable properties of an emulator:

- it evaluates quickly;

- it is expressive enough to provide good approximations to the simulator and to allow meaningful prior judgements;

- it predicts that $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$ are very close when $\boldsymbol{x}$ and $\boldsymbol{x}'$ are very close.

The general approach is to propose a stochastic form for $f(\boldsymbol{x})$ and then consider the properties of this function when conditioned to agree with the results of the simulator for the parameter values $\boldsymbol{x}_i$ at which the simulator has been run. The effectiveness of the emulator will depend on the choice of the emulator's form, which in general will be chosen using judgement, exploration of the data and tuning. A typical choice for an emulator for a scalar-valued $f(\boldsymbol{x})$ is

$$f(\boldsymbol{x}) = \sum_i \beta_i g_i(\boldsymbol{x}) + u(\boldsymbol{x}), \tag{8}$$

or, for a vector-valued $\boldsymbol{f}(\boldsymbol{x})$,

$$f_i(\boldsymbol{x}) = \sum_j \beta_{ij} g_{ij}(\boldsymbol{x}) + u_i(\boldsymbol{x}).$$

For the rest of this section, attention is restricted to scalar-valued f for simplicity of notation. For the application to NAME, f is vector-valued but this is handled by constructing separate scalar emulators for each f_i.

Here, $g_i(\boldsymbol{x})$ is chosen to be simple functions (for instance polynomials), and the β_i are unknown coefficients. These terms control the global trend of the model. The function $u(\boldsymbol{x})$ controls the local variation of the model. Typically, it is supposed that the expected value of u is zero, i.e. $E(u(\boldsymbol{x})) = 0$, and that the correlation, $\mathrm{Corr}(u(\boldsymbol{x}), u(\boldsymbol{x}'))$, is some function of the distance between $\boldsymbol{x}$ and $\boldsymbol{x}'$, such that the correlation falls as parameters get further apart. For example, a popular choice and the one used for this application is

$$\mathrm{Corr}(u(\boldsymbol{x}_1), u(\boldsymbol{x}_2)) = \exp\left(-\left(\frac{d(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\delta}\right)^2\right),$$

where $d(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the Euclidean distance between the parameters, and δ is the "correlation length", a parameter that determines how quickly correlation falls with distance. Finally, it is commonly assumed that the variance of u, $\mathrm{Var}(u(\boldsymbol{x}))$, equals σ^2 for all $\boldsymbol{x}$, so the variance of the local term is constant across the parameter space.

Building an emulator therefore involves using a collection of simulator runs $f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_n)$ to

- identify the basis functions g_i,

- estimate the β_i, and

- fit the residual function $u(\boldsymbol{x})$.

Such an emulator then provides predictions for $f(\boldsymbol{x})$ at a new $\boldsymbol{x}$. Since it is a statistical model, this prediction also comes with an associated uncertainty, which will be low near observed simulator runs and higher away from them. In the Bayesian linear framework used in this paper, the prediction

takes the form of the expected value of f given the results of the collection of simulator runs, and the associated uncertainty is the variance of f given the simulator runs; see Sect. 5.2 for full details. Figure 4 shows an emulator for a scalar-valued function of one variable.

Computer simulator applications often involve a mixture of observed simulator runs and expert knowledge, making a Bayesian framework a natural choice to build emulators. However, specification of a full joint probability distribution for the problem is difficult and often leads to computational challenges. With enough simulator evaluations, a successful method for fitting emulators has been to use a standard (non-Bayesian) least-square regression (that is, with no prior) to estimate the β and use the residual variance from the regression for σ^2. This is then used in a Bayesian analysis of u. These results should be very similar to a Bayesian analysis of both β and u, without needing to worry about the prior judgements for β.

In this application, there are enough evaluations to build an emulator for the *fast* simulator by this method. However, it is the slow simulator that is really of interest. A method proposed in Cumming and Goldstein (2009) is applied, in which the emulators for the fast and slow simulators are linked through a simple Bayesian model, in which the *form* of the emulators are the same but the coefficients and parameters are different. Even in this approach, a full Bayesian calculation is computationally demanding and in high dimensions can be very sensitive to the initial prior specifications. Further, specifying the full high-dimensional probability distributions that properly reflect expert judgements is an extremely difficult task. In this paper, the alternative Bayes linear approach (Goldstein and Wooff, 2007) is used instead, in the analysis of the residual function for the fast simulator, and also the analysis of the link between the fast and slow emulators. The next section gives a brief description of the Bayes linear approach, before the specific model used to link the two emulators is introduced.

5.2 Bayes linear methods

As with a full Bayes analysis, the Bayes linear analysis combines prior judgements with observations (of the simulator) through simple equations. Bayes linear analysis does not, however, require expert judgements to be specified as a full joint prior probability distribution for all variables. Rather, the experts need only to specify expectations, variances and covariances for a few relevant quantities. Similarly, rather than a joint posterior probability distribution, Bayes linear analysis leads to adjusted expectations, variances and covariances for relevant quantities. Given a vector of data $\boldsymbol{D}$ (for example, simulator runs $f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_n)$ that have been evaluated), the representation of f in Eq. (8) and a vector of quantities of interest $\boldsymbol{B}$ (for example, the value of the simulator $f(\boldsymbol{x})$ at some new $\boldsymbol{x}$ at which the simulator has not yet
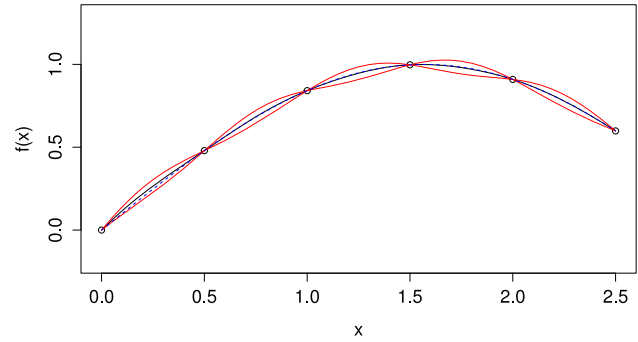
Figure 4. One-dimensional example of an emulator. The points represent the six evaluations of $f(\boldsymbol{x})$, the black line is the emulator's prediction, and the red lines give 2 standard deviations. The blue dashed line is the true value of $f(\boldsymbol{x})$.

been evaluated), the adjusted expectation and variance for B are given by

$$E_{\mathrm{D}}(B) = E(B) + \mathrm{Cov}(B, D)\mathrm{Var}(D)^{-1}(D - E(D)), \quad (9)$$

$$\mathrm{Var}_{\mathrm{D}}(B) = \mathrm{Var}(B) - \mathrm{Cov}(B, D)\mathrm{Var}(D)^{-1}\mathrm{Cov}(D, B). \quad (10)$$

Note that these equations hold for arbitrary D, not just the D defined in the previous section (the set of slow simulator outputs). In particular, we will often replace D with D' (the set of fast simulator outputs) in these equations.

If

$$f'(\boldsymbol{x}) = \sum_{i=1}^{p} \beta'_i g_i(\boldsymbol{x}) + u'(\boldsymbol{x}) \quad (11)$$

is an emulator for the fast simulator, then a Bayes linear adjustment requires prior expectations and variances for the β'_i and $u'(\boldsymbol{x})$, and all covariances between these components to be specified. The quantities are needed to compute the prior quantities $E(B)$, $E(D)$, $\mathrm{Var}(D)$, $\mathrm{Var}(B)$ and $\mathrm{Cov}(B, D)$. For example, if B is a new $f'(\boldsymbol{x})$ and D are the results of the fast simulator runs, we need to know

$$E(B) = \sum_{i=1}^{p} E\left(\beta'_i\right) g_i(\boldsymbol{x}) + E(u'(\boldsymbol{x})),$$

$$\mathrm{Var}(B) = \mathrm{Var}\left(\sum_{i=1}^{p} \beta'_i g_i(\boldsymbol{x}) + u'(\boldsymbol{x})\right),$$

and so on.

Then, given the results D, the adjusted quantities $E_{\mathrm{D}}(B)$ and $\mathrm{Var}_{\mathrm{D}}(B)$ can be computed from Eqs. (9) and (10).

As explained in the previous section, this study involves the further simplification of replacing the Bayes linear determination of some of the quantities with estimates from a least-square regression. The β'_i are replaced by the least-square regression estimates $\hat{\beta}'_i$ (this may be thought of as taking $E(\beta'_i) = \hat{\beta}'_i$ with zero variance), $E(u'(\boldsymbol{x}))$ is taken to be

zero and $\text{Var}(u'(\boldsymbol{x}))$ is taken to be the residual variance from the regression. The only remaining component needed for the Bayes linear adjustment in Eqs. (9) and (10) is the correlation $\text{Corr}(u'(\boldsymbol{x}_1), u'(\boldsymbol{x}_2))$. This can be fit using various methods; more details of this can be found in Appendix A1. Once this is done, Eqs. (9) and (10) can be used to calculate the adjusted expectation and variance for any new $f'(\boldsymbol{x})$ given a collection of previous fast simulator runs. This expectation and variance give a prediction and uncertainty quantification for the behaviour of the fast simulator at this new $\boldsymbol{x}$.

5.3 Linking fast and slow simulators

Having built an emulator for the fast simulator in Eq. (11) using the Bayes linear approach, the next step is to link this to an emulator for $f(\boldsymbol{x})$ from Eq. (8). This involves building a model for the relationship between the two emulators.

Such a model will typically have some unknown parameters, and the Bayes linear approach can be used again to learn about these parameters and hence provide an adjusted expectation and variance for $f(\boldsymbol{x})$, given a collection of fast simulator evaluations and a collection of slow simulator evaluations. Notice that in Eqs. (8) and (11), the basis functions $g_i(x)$ are the same in both emulators. That is, it is supposed that the mean trend of the fast simulator f' has the same form (but different coefficients) as the simulator of interest f. If f' is a reasonable approximation for f (for instance, an older version of f or a version of f run at lower resolution) this supposition will usually be valid.

Further, the coefficients β_i and β_i' will often be similar. A model linking these coefficients will allow the fast simulator runs to provide information about the β_i. At the same time, this model must be flexible enough that it does not impose a strong link where none exists. The same can be said of the link between $u(\boldsymbol{x})$ and $u'(\boldsymbol{x})$. A simple model is

$$\beta_i = \rho_i \beta_i' + c_i$$
$$u(\boldsymbol{x}) = \rho_0 u'(\boldsymbol{x}) + w(\boldsymbol{x}),$$

where ρ_0, ρ_i are unknown multipliers and c_i are unknown scalars. If the two simulators are very similar, then most ρ_i will be near 1 and most c_i will be near 0. If the value of $g_i(\boldsymbol{x})$ has a much smaller effect on the fast emulator that on the slow emulator, ρ_i will be much larger than 1. Where the value of $g_i(\boldsymbol{x})$ has a much larger effect on the fast emulator that on the slow emulator, ρ_i will be near zero. If $g_i(\boldsymbol{x})$ has an opposite effect on the fast emulator and the slow emulator, then ρ_i will be negative. The emulation process therefore involves using the fast simulator to work out the form of the emulator, to estimate the β_i' and make inferences about u' (via a Bayes linear adjustment), and then using the slow simulator to make inferences about the ρ_i and w (via another Bayes linear adjustment). Note that underlying this approach is the assumption that the slow simulator runs do not provide any more information about the fast simulator.

In this application, it turned out that this could be further simplified to

$$\beta_i = \rho_i \beta_i'$$
$$u(\boldsymbol{x}) = \rho_0 u'(\boldsymbol{x}) + w(\boldsymbol{x}) \qquad (12)$$

without noticeably reducing the effectiveness of the emulators.

This model requires prior expectations, variances and covariances for the ρ_i and ρ_0, as well as for $w(\boldsymbol{x})$. In Appendix A1.1, more details of these prior requirements are provided.

With these choices, the Bayes linear adjustment for a new $f(\boldsymbol{x})$ can now be performed. This calculation and the resulting equations are somewhat technical, and so they are given in Appendix A2; in particular the adjusted expectation and variance for f can be found in Eqs. (A2) and (A3).

5.4 Diagnostics and validation

It is important to check that an emulator is performing well before using it to make predictions. There are several possible reasons an emulator would be poor. The form of the mean function (that is, the component $\sum \beta_i g_i(\boldsymbol{x})$) could be missing an important term or even be totally misguided. The form of the correlation function might be inappropriate. The parameters in the correlation function (in this application, the correlation length) could be set at inappropriate values. Finally, some other assumptions, such as the assumption that $\text{Var}(u(\boldsymbol{x}))$ is the same for all $\boldsymbol{x}$, could be seriously misleading.

The mean function plays a large role in these emulators. The usual diagnostics from linear models can be valuable in assessing the adequacy of the chosen mean function.

The coefficient of determination, R^2, a statistic that represents the proportion of variation explained by the parameters in the least-square regression, is a useful number to check first. If this is low, then the mean function is not explaining much of the variation in the simulator output, and adding new terms or changing the form of the mean function entirely should be considered. Examining the residuals can also be useful in this process, in particular whether there are regions of the parameter space where the residuals are systematically large in one direction.

A simple and effective method of validation is leave-one-out validation. In this procedure, all but one of the observed simulator runs are used to build an emulator, and this emulator is used to predict the one run that was left out. For n simulator runs, this gives n emulators and predictions. If the emulators frequently predict the left-out values to be far from the observed simulator run, this suggests a problem with the emulator. Here, "far from" means relative to the variance of the emulator – a useful rule of thumb is that about 95 % of the validation runs should be within 3 standard deviations of the prediction.

Table 4. Number of outputs for which each parameter was judged active (and hence included in the emulator for that output). Recall that x_7 and x_8 are linked, and so x_8 is not present in the table, and similarly for x_{10} which is linked to x_9. Also x_2 is used only for the thin-layer source results.

Parameter	x_1	x_3	x_4	x_5	x_6	x_7	x_9	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}
Times active	75	75	18	18	0	61	15	4	58	0	1	0	0	0	0

If this proportion of successful prediction is much lower than 95 %, this might signal a fundamental problem with the mean function and/or the form of the correlation function, but it can often simply signal a poor choice of correlation length. If the correlation length is too high, then the emulator variance will be too low and hence many observations of the simulator will be judged "too far" from the emulator predictions. In contrast, if the correlation length is too low, then the emulator will not be able to capture many patterns of local variation from the mean function that may be present (specifically, any such patterns that exist over distances much higher than the correlation length). It is often possible to tune the correlation length so that the proportion of successful validations is around 95 %.

6 Application to NAME

6.1 Choosing basis functions and eliminating inactive parameters

We first consider the choice of basis functions $g_i(\boldsymbol{x})$. This involves choosing the form of the functions and also which parameters are even used. Often, some parameters have negligible impact on some simulator outputs and therefore removing them from the corresponding emulator entirely is advisable. Such parameters are said to be "inactive" in this emulator, whereas the parameters that appear in the emulator are called "active".

Inactive parameters can be identified by stepwise selection using criteria such as adjusted R^2 (an adjustment to the usual coefficient of determination that takes into account the number of terms in the regression, or other popular criteria such as the Akaike information criterion or the Bayesian information criterion. In this application, a parameter was judged inactive if removing it led to little reduction in the adjusted R^2 and if the corresponding leave-one-out validation did not perform appreciably worse with it excluded. The details for this process are included in Appendix A3.1. Note that active variables are not necessarily extremely important parameters; they simply provide some information that would be lost by excluding them.

The result was that the chosen g_i were quadratic and lower-order terms, i.e.

$$f'(\boldsymbol{x}) = \sum_i a_i x_i^2 + \sum_i \sum_{j \neq i} b_{ij} x_i x_j + \sum_i c_i x_i + u'(\boldsymbol{x}).$$

For most output quantities, this led to the emulators with four active variables, with more in a few of the 75 output areas. Parameters x_1 (plume height) and x_3 (mass eruption rate) were active in all models, with x_7 (standard deviation of free tropospheric turbulence) and x_{12} (precipitation rate required for wet deposition) active in most. Parameters x_6 (ash density), x_{13} (scavenging coefficient parameter A for rain) and x_{15}–x_{18} (scavenging coefficient B and dry deposition resistances) were active in no emulators. A summary of the number of times each parameter was active is shown in Table 4.

6.2 Emulating the fast simulator

Least-square regression was used to fit the above linear models. Each of these linear models now gives an estimate for β_i' and a residual variance that can be used for $\mathrm{Var}(u'(\boldsymbol{x}))$. This was used to build emulators for the fast simulator, again using standard methods. Details of the choices made and justifications for these can be found in Appendix A3.2. Validation was performed by attempting to predict the 200 observations for the simulator from $\mathcal{X}_\mathrm{S}$ using the remaining 1500; the proportion of $f'(\mathcal{X}_\mathrm{S})$ predicted reliably (that is, within 3 standard deviations of the emulator variance) for each output ranged from 94.5 to 99 %. An example of validation can be seen in Fig. 5. This suggests that the emulator is a useful tool for prediction.

The next step is to link the fast simulator to the slow simulator and use the runs $f(\mathcal{X}_\mathrm{S})$ to make predictions for the slow simulator.

6.3 Emulating the slow simulator

The emulator for the fast simulator is linked to that of the slow simulator through Eq. (12) (recall that the emulators for the slow and fast simulators are given by Eqs. 8 and 11, respectively). This requires prior judgements for ρ_i and $w(\boldsymbol{x})$. For the latter, the judgements used were that $E(w(\boldsymbol{x})) = 0$, $\mathrm{Var}(w(\boldsymbol{x})) = \mathrm{Var}(u'(\boldsymbol{x}))$ and the correlation structure is the same form as that of $u'(\boldsymbol{x})$. This correlation structure was tuned in the same way as for the fast emulator (see Appendix A3.2). For the slow simulator, it was found that longer correlations lengths were more appropriate, with values ranging from 0.2 to 0.4. Expectations, variances and covariances for ρ_i were specified using the least-square method in Appendix A1.1.

With this model, the adjusted expectation and variance $E_\mathrm{D}(f(\boldsymbol{x}))$ and $\mathrm{Var}_\mathrm{D} f(\boldsymbol{x}))$ can be computed for any new $\boldsymbol{x}$
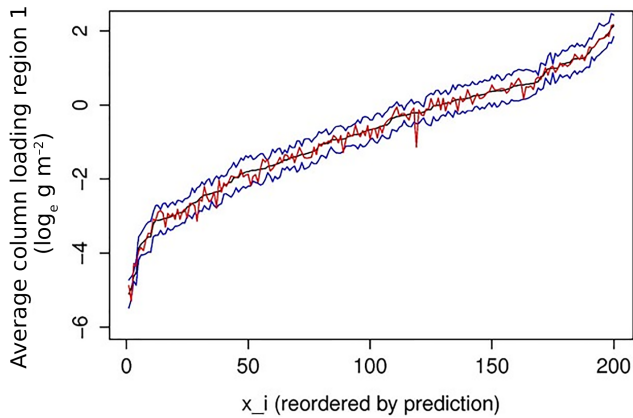
Figure 5. Validation plot for the emulator (for the fast simulator) of the first output. Emulator expected value for the parameter sets in $\mathcal{X}_S$ is shown in black, with an interval of 3 standard deviations each side shown in blue. The red line shows the true simulator output at each parameter set. The parameters have been ordered from lowest to highest emulator prediction.

using Eqs. (A2) and (A3) in Appendix A2. Note that this calculation includes the adjusted expectation and variance of the ρ_i. Examining these quantities shows which regions and for which g_i the differences between the fast and slow simulators are most pronounced. In conjunction with the β_i', they also give more insight into how the active parameters drive the simulator output.

Validation followed a similar method to that for the fast emulator. In this case, over the 75 regions, the proportion of successful predictions from the validation again ranged from 94.5 to 99 %.

For most emulators, the ρ_i were close to 1 (typically between 0.95 and 1.05) for all β_i. With the difference between the fast and slow simulators being only a factor of $\sqrt{10}$ in the simulation noise and with the simulation noise being kept low by averaging over large regions, this is perhaps expected. The main exceptions were regions where the fast simulator predicted relatively little ash compared with the slow simulator – in these cases the ρ_i were typically between 0.5 and 0.75 systematically (that is, no particular parameter was affected more than others). In no case did a ρ_i approach 0 (which would indicate a parameter becoming inactive in the slow emulator) or change sign. The only multiplier that was frequently low was ρ_0, the multiplier for the residual process. In conclusion, the link between mean functions of the two emulators is strong and consistent, in the sense that either the ρ are all near 1, or they are all near some α so that the difference is mostly a rescaling. The local variations, in contrast, are usually unrelated, with ρ_0 near zero. This suggests that the fast simulator could be used more extensively in future applications significantly reducing simulation run times. Given the similarity between the two simulators, it is not surprising that the multi-level emulation method works

smoothly in this application; in applications with more fundamental structural differences between the simulators, it is likely that more careful modelling of the link would be required.

6.4 Implications for NAME case study

The adjusted β_i confirm that the simulator behaves broadly as one would expect. As MER increases (either due to its dependence on the plume rise height, x_1 via Eq. 1 or alterations in 1 caused by x_3) the quantity of ash in the atmosphere increases. When the precipitation threshold is higher, higher values of ash in the atmosphere are also predicted. This is due to less ash being deposited to the surface as only precipitation rates above the threshold lead to wet deposition. When the PSD favours large particles, predicted airborne ash reduces because these heavy particles sediment much more quickly than small particles and therefore are removed from the atmosphere and not available for long-range transport. Interpretation of the other parameters in the simulation is more complicated, because their interaction terms make it impossible to state that increasing them will consistently move the output in one direction or another.

Of all the parameters, the plume height drives the output most strongly, followed by the MER and the precipitation threshold. In all regions, the β_i with the highest estimate corresponded to a function of the plume height, x_1 (either the x_1 term or the x_1^2 term). Despite this, the impact of some of the other parameters is not negligible. Table 5 contains averages of the expected values of some of the β_i across all the regions in which the corresponding parameters were active. The terms chosen for this table were those with the largest expected values. It can be seen that x_1 is certainly the dominant parameter but others are still influential. Note that the appearance of x_{11} in this table should be viewed with caution: it was only active in a few regions, and one of those (region 63) had a very unusual emulator with particularly high β_i. It is also interesting to note that x_4 and x_5 do not appear in this table: although active parameters in many regions, their effects are consistently small. These parameters refine the emulator slightly, but not nearly as much as the main ones. Finally, x_9 was rarely active and has a similar physical effect to x_7.

Interactions between the parameters (that is, the terms of the form $\beta_{ij} x_i x_j$) were small for most i, j pairs but, as can be seen in Table 5, there are some notable exceptions. Each pair of x_1, x_7 and x_{12} (plume height, turbulence in the free troposphere and precipitation threshold, respectively) have negative interaction terms, with all such interactions relatively strong except for the $x_7 x_{12}$ pair.

The presence of these interactions makes interpretation of the parameters' impact more difficult. For example, ignoring the interactions, it would appear that increasing plume height increases column loading and increasing the precipitation threshold increases column loading. Typically this is true, but the presence of the negative interaction term means,
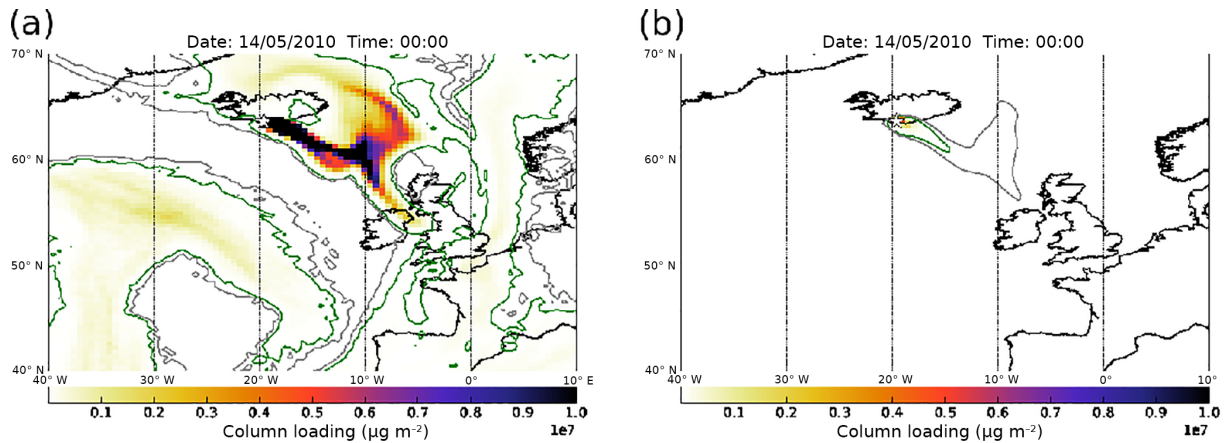
Figure 6. NAME ash column loading for parameter choices with the highest and lowest expected ash column loadings in the first geographical region at 00:00 UTC on 14 May. The contours are as in Fig. 2.

firstly, that the magnitude of these increases will not be as large as one might first expect and, secondly, that there are some (x_1, x_{12}) pairs where increasing x_{12} can actually reduce the output. This is particularly pronounced with parameter x_7, where the magnitudes of its interaction terms tend to be much larger than the x_7 and x_7^2 terms – it is not possible to make simple statements about whether increasing x_7 will increase or decrease the NAME output. Finally, whenever both x_7 and x_9 (standard deviation of turbulence and Lagrangian timescale for turbulence) were active together, they exhibited very strong negative interaction terms (the highest interactions seen anywhere in the analysis, apart from the possibly spurious x_{11} ones). Since x_9 was rarely active and is related to x_7 physically, this suggests two possible refinements: either forcing x_9 to be active whenever x_7 is, since there is a strong relationship between them, or using a single parameter to represent the range of possible diffusivities $K_u = \sigma_u^2 \tau_u$ like we have for unresolved mesoscale motions (see Sect. 3.2.3). Note x_7 represents both horizontal and vertical turbulence because it is linked to x_8.

It should be pointed out that the above conclusions are not based directly on (and cannot be made from) the values in Table 5, since these are just averages. The behaviour above was observed by exploring the effects of moving around the parameter space for each emulator separately. Given the number of regions and parameters, it is not possible to provide the full results of this exploration, but this explains why using the averages alone cannot reproduce some of our conclusions.

The emulators provide insight into which areas of the parameter space will lead to high values of simulated ash column loading and which areas will lead to low values of ash column loading. As an extreme case, the parameters giving the lowest and highest predictions of ash column loading can be identified. This was done for the first hour of 14 May, giving two parameter sets at which the simulator was evaluated. The results of these simulator evaluations can be seen in

Table 5. Average of the expected values of selected β coefficients across all regions. Any term not present has small coefficients in all regions or was inactive in all regions. Note that the sensitivity to x_{11} is overstated because it was active only in a few regions, and one of those regions exhibited unusual behaviour.

Linear model term	Average
x_1^2	−2.26
x_7^2	0.55
x_{11}^2	−1.45
x_{12}^2	−0.73
x_1	8.05
x_3	2.21
x_{11}	8.25
x_{12}	2.11
$x_1 x_7$	−0.93
$x_1 x_{11}$	−3.73
$x_1 x_{12}$	−0.91
$x_7 x_9$	−1.05
$x_7 x_{11}$	−3.55
$x_7 x_{12}$	−0.12

Fig. 6. This gives an idea of the range of possibilities admitted by the expert judgements from Sect. 3. As can be seen, these two plots are very different; that is, the ranges in Table 1 cover a broad range of simulator behaviour. Note however that our choice of parameter ranges has deliberately tried to cover the whole range of possible values and that, for the parameters relating to turbulence, it is not plausible that the maximum values could be present throughout the whole atmosphere.

This is a function of the current NAME parameterisation of free tropospheric turbulence (i.e. the fact that NAME uses the same parameter everywhere).

Now, only a small region of this parameter space will lead to simulations that resemble the observations on this day.

The emulators can be used to identify this region of parameter space. Since emulators can be evaluated very quickly, predictions and their associated uncertainty can be generated for very many candidate parameters, and all predictions that are very far from the observations can be rejected. This procedure, called "history matching", focuses on the plausible regions of parameter space and allows more accurate emulators to be built within them. This is because in a reduced parameter space, the form of the emulator can be changed to better model the behaviour in that subspace, without being concerned about global behaviour. In such a region, previously inactive parameters may once again become active, and more illuminating insights can be made. Performing this analysis for NAME is beyond the scope of this paper, but will be covered in a second study.

7 Conclusions

In this paper it has been shown that a Bayes linear emulation approach can be used to identify source and internal model parameters that contribute most to the uncertainty in the long-range transport of volcanic ash in a complex VATD simulator. The approach presented is applicable to other complex simulators that have long computation times and many parameters contributing to the overall prediction uncertainty. This approach uses latin hypercube sampling of the plausible parameter ranges determined through expert elicitation. All parameters are varied in each simulator run and therefore information about the importance of the parameters and their interaction can be investigated simultaneously. This gives a much more realistic estimate of the uncertainty than using one-at-a-time tests and provides much more useful information to model developers and those planning observational campaigns.

Here 1700 simulator runs have been used to build 75 emulators representing the average ash column loading in regions on 14 May 2010. These simulator evaluations comprised 1500 fast simulator runs and 200 slow simulator runs. The analysis demonstrated the strength of using approximate simulators to determine the general trend of a simulator and provide plausible priors, before using a relatively small number of accurate simulator runs to refine the emulator. Bayes linear methods were used to reduce computational complexity and the need for detailed prior judgements that we may not be certain of.

For this case the most important parameters are plume height, mass eruption rate, free troposphere turbulence levels and precipitation threshold for wet deposition. There is also a strong negative relationship between plume height and free troposphere turbulence and between plume height and precipitation threshold. This means that, for example, although increasing these parameters individually typically increases column loading, increasing both parameters at the same time does not increase column loading as much as would be expected looking only at the individual parameters. These conclusions should be tested in other situations to assess how widely they hold. An assessment of the impact of meteorological uncertainty is also required but this is beyond the scope of this study. This information can be used to inform future research priorities (e.g. the addition of a more complex free tropospheric turbulence scheme which varies spatially and temporally (see Dacre et al., 2015) and investigating the importance of the precipitation threshold within the NAME simulator) and observational capabilities (e.g. a mobile radar to observe plume height) and measurement campaigns (e.g. in situ observations of ash PSD). Furthermore, this analysis can be used to prioritise variables to perturb in a small operational ensemble.

This study has shown the range of possible ash column loading distributions possible from sampling the parameter space determined by the ranges elicited from simulator experts. Only a small region of this parameter space will lead to simulations that resemble the observations on this day. Emulators can be used to identify this region of parameter space as they can be evaluated very quickly. The resulting predictions and their associated uncertainty can be generated for very many candidate parameters, and all predictions that are very far from the observations can be rejected. This procedure, known as "history matching", focuses on the plausible regions and allows more accurate emulators to be built within them. This analysis is beyond the scope of this paper. This will form the basis of a future study but could further inform the parameter perturbations used in an operational ensemble. The approach presented here could be easily applied to other case studies, simulators or hazards. Furthermore, an ensemble of emulator evaluations could be used to produce probabilistic hazard forecasts.

Appendix A: Adjusting slow emulators using fast emulators

A1   Fitting the fast emulator

Recall that for a Bayes linear calculation, one needs prior specifications of expectations, variances and correlations of all unknown quantities. For the fast emulator, the β_i' are fixed at their least-square estimates, $E(u'(\boldsymbol{x}))$ is taken to be zero, and $\mathrm{Var}(u'(\boldsymbol{x}))$ is set to the residual standard deviation. This leaves only $\mathrm{Corr}(u'(\boldsymbol{x}_1), u'(\boldsymbol{x}_2))$ to be estimated. A typical approach is to specify a correlation function that depends only on $d(\boldsymbol{x}_1, \boldsymbol{x}_2)$, the distance between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. A common choice, used in this study, is

$$\mathrm{Corr}\left(u'(\boldsymbol{x}_1), u'(\boldsymbol{x}_2)\right) = \exp\left(-\left(\frac{d(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\delta'}\right)^2\right),$$

although other choices are possible; in particular using a different correlation length δ' for each direction would often be useful, although did not prove necessary in this application.

The parameter δ' governs the strength of the correlation and must be estimated from the observed residuals by some method. A formal estimation can be performed using the variogram methods of Cressie (1993) applied for example in Cumming and Goldstein (2009). A more heuristic approach has been successful in many other applications (Vernon et al., 2010; Goldstein et al., 2010; Goldstein and Huntley, 2017). This involved the argument that, for a polynomial mean function, a plausible value of δ' is $\frac{1}{p+1}$, where p is the highest-order term in the polynomial fit. This starting value can then be explored and adjusted "by hand". A popular strategy is a leave-one-out exploration: for each parameter $\boldsymbol{x}_i$, calculate the adjusted expectation and variance for $f(\boldsymbol{x}_i)$ using all the other $\boldsymbol{x}_j$ and a trial value of δ'. The observed value of $f(\boldsymbol{x}_i)$ can then be used to see whether the prediction was accurate or not. The value of δ' used can be adjusted to balance two competing requirements: that most of the predictions are close (relative to the adjusted variance) to the observed values and that the variances are small. For example, if many more than 5 % of predictions are more than 3 standard deviations away from the observation, δ' is unlikely to be a good choice, so a good value of δ' should satisfy this requirement while keeping the variances as low as possible.

A1.1   Prior judgements for linked emulators

Using the linking model in Eq. (12), an adjustment of the slow emulator involves prior expectations, variances, and covariances for ρ_i, ρ_0 and $w(\boldsymbol{x})$. A simple approach is to use

$$E(\rho_i) = 1$$
$$\mathrm{Var}(\rho_i) = \sigma_\rho^2$$
$$\mathrm{Cov}(\rho_i, \rho_j) = r,$$

reducing the specification for the multiplier to two numbers σ_ρ^2 and r. Note that ρ_0 is included in the above specifications. This leaves only $w(\boldsymbol{x})$ to consider. A natural choice is to use the same form as is used for $u'(\boldsymbol{x})$, including the same variance and the same correlation length δ. Another option is to use the same correlation structure but allow $\mathrm{Var}(w(\boldsymbol{x})) = \sigma_w^2$ to be different from $\mathrm{Var}(u'(\boldsymbol{x}))$. Finally, a very useful simplification is to take $\mathrm{Corr}(w(\boldsymbol{x}), \rho_i) = 0$ for all i (including 0).

Thus, the link between f' and f is provided by $\tau = \{\sigma_\rho^2, r, \sigma_w^2\}$ – only these three values need to be specified now (or only the first two, depending on earlier choices). Were it possible to specify values for τ, this would provide all the ingredients to perform a Bayes linear calculation to learn about the slow simulator using the (adjusted) fast emulator and the evaluations $f(\mathcal{X}_S)$. However, the quantities in τ are difficult to think about, so expecting an expert to be able to specify them is unrealistic.

Instead, plausible values for τ can be generated using the differences $d(\boldsymbol{x}) = f(\boldsymbol{x}) - f'(\boldsymbol{x})$ for each $\boldsymbol{x} \in \mathcal{X}_S$. As calculated in Cumming and Goldstein (2009),

$$\mathrm{Var}(d(\boldsymbol{x})) = \sigma_\rho^2 \phi(\boldsymbol{x}) + \sigma_\rho^2 r \psi(\boldsymbol{x}) + \sigma_w^2, \qquad (A1)$$

where

$$\phi(\boldsymbol{x}) = \sum_{i=1}^{p+1} b_i(\boldsymbol{x})^2$$
$$\psi(\boldsymbol{x}) = \sum_{i \neq j} b_i(\boldsymbol{x}) b_j(\boldsymbol{x}),$$

with

$$b(\boldsymbol{x}) = \left(\beta_1' g_1(\boldsymbol{x}), \ldots, \beta_p'(\boldsymbol{x}) g_p(\boldsymbol{x}), u'(\boldsymbol{x})\right),$$

noting that $u'(\boldsymbol{x})$ is known for each $\boldsymbol{x} \in \mathcal{X}_S$ because the fast simulator was evaluated at each such point. Further, $E(d(\boldsymbol{x})) = 0$, and hence $\mathrm{Var}(d(\boldsymbol{x})) = E(d(\boldsymbol{x})^2)$ and so, from Eq. (A1),

$$E\left(d(\boldsymbol{x})^2\right) = \sigma_\rho^2 \phi(\boldsymbol{x}) + \sigma_\rho^2 r \psi(\boldsymbol{x}) + \sigma_w^2,$$

and for $\boldsymbol{x} \in \mathcal{X}_S$ everything on the right-hand side of this equation is known except for τ. Replacing $E(d(\boldsymbol{x})^2)$ with the observed $d(\boldsymbol{x})^2$, this gives $|\mathcal{X}_S|$ (in our application, 200) linear equations in three unknowns, and a least-square fit can be used to estimate these three unknowns and hence $\hat{\tau}$. This $\hat{\tau}$ can then be used as the prior judgements for the link between the emulators. Note that this approach works only because both fast and slow simulators are evaluated at $\mathcal{X}_S$.

A2   Adjusting the slow emulator

Suppose an emulator f' has been constructed as in Eq. (11) by using D'; in particular we suppose that the β_i' are known and that u' has had its mean and variance adjusted using

Eqs. (9) and (10) (with D replaced by D'). We also assume the link Eq. (12) between the fast and slow emulators and that priors have been specified for ρ_i and w, for instance by the methods in Appendix A1.1. The adjusted fast emulator and the slow simulator runs D are available to be used in the adjustment of ρ and w, and hence the adjustment of $f(\boldsymbol{x})$ for any new $\boldsymbol{x}$.

First, we have

$$E_{\mathrm{D}}(\rho_i) = 1 + \mathrm{Cov}(\rho_i, D)\,\mathrm{Var}(D)^{-1}(D - E(D)).$$

The prior expectation for each element of D is simply the value observed for the corresponding element of D'. Also,

$$\begin{aligned}
\mathrm{Cov}(\rho_i, D_j) &= \mathrm{Cov}\left(\rho_i, \sum_k \rho_k \beta_k' g_k(\boldsymbol{x}_j) + \rho_0 u'(\boldsymbol{x}_j) + w(\boldsymbol{x}_j)\right) \\
&= \sum_k \mathrm{Cov}(\rho_i, \rho_k)\,\beta_k' g_k(\boldsymbol{x}_j) + \mathrm{Cov}(\rho_i, \rho_0) u'(\boldsymbol{x}_j) \\
&= \sum_{k=1}^{p+1} \mathrm{Cov}(\rho_i, \rho_k)\,b(\boldsymbol{x}_j)_k.
\end{aligned}$$

Finally, the variance matrix $\mathbf{Var}(D)$ is built from elements of the form

$$\begin{aligned}
\mathrm{Cov}(D_1, D_2) &= \mathrm{Cov}\left(\sum_i \rho_i \beta_i' g_i(\boldsymbol{x}_1) + \rho_0 u'(\boldsymbol{x}_1) + w(\boldsymbol{x}_1),\right. \\
&\qquad \left. \sum_i \rho_i \beta_i' g_i(\boldsymbol{x}_2) + \rho_0 u'(\boldsymbol{x}_2) + w(\boldsymbol{x}_2)\right) \\
&= \sum_{i,j} \mathrm{Cov}(\rho_i, \rho_j)\,b(\boldsymbol{x}_1)\,b(\boldsymbol{x}_2) \\
&\quad + \mathrm{Cov}(w(\boldsymbol{x}_1), w(\boldsymbol{x}_2)).
\end{aligned}$$

Recall in the above that the fast simulator has been evaluated at both $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, so in particular $u'(\boldsymbol{x}_i)$ are known. This is all that is needed to calculate $E_{\mathrm{D}}(\rho_i)$.

The adjusted variance for ρ is given by

$$\mathbf{Var}_{\mathrm{D}}(\rho) = \mathrm{Var}(\rho) - \mathrm{Cov}(\rho, D)\,\mathrm{Var}(D)^{-1}\mathrm{Cov}(D, \rho),$$

which can be calculated from the expressions above.

The adjustment for the residual $w(\boldsymbol{x})$ is simpler:

$$E_{\mathrm{D}}(w(\boldsymbol{x})) = \mathrm{Cov}(w(\boldsymbol{x}), D)\,\mathrm{Var}(D)^{-1}(D - E(D))$$

$$\begin{aligned}
\mathbf{Var}_{\mathrm{D}}(w(\boldsymbol{x})) &= \mathrm{Var}(w(\boldsymbol{x})) - \mathrm{Cov}(w(\boldsymbol{x}), D)\,\mathrm{Var}(D)^{-1} \\
&\qquad \mathrm{Cov}(D, w(\boldsymbol{x})),
\end{aligned}$$

where

$$\mathrm{Cov}(w(\boldsymbol{x}), D_i) = \mathrm{Cov}(w(\boldsymbol{x}), w(\boldsymbol{x}_i)).$$

Then, for any $\boldsymbol{x}$ such that $\boldsymbol{x} \in \mathcal{X}_{\mathrm{F}}$ (the parameters used for the fast but not slow simulator runs), Cumming and Goldstein

(2009) showed that the Bayes linear adjustment for $f(\boldsymbol{x})$ is given by

$$E_{\mathrm{D}}(f(\boldsymbol{x})) = b(\boldsymbol{x})^T E_{\mathrm{D}}(\rho) + E_{\mathrm{D}}(w(\boldsymbol{x})) \tag{A2}$$

$$\begin{aligned}
\mathbf{Var}_{\mathrm{D}}(f(\boldsymbol{x})) &= b(\boldsymbol{x})^T \mathrm{Var}_{\mathrm{D}}(\rho)b(\boldsymbol{x}) + \mathbf{Var}_{\mathrm{D}}(w(\boldsymbol{x})) \\
&\quad + 2b(\boldsymbol{x})\mathrm{Cov}_{\mathrm{D}}(\rho, w(\boldsymbol{x})), \tag{A3}
\end{aligned}$$

where

$$\begin{aligned}
\mathrm{Cov}_{\mathrm{D}}(\rho, w(\boldsymbol{x})) &= \mathrm{Cov}(\rho, w(\boldsymbol{x})) - \mathrm{Cov}(\rho, D)\mathrm{Var}(D)^{-1} \\
&\qquad \mathrm{Cov}(D, w(\boldsymbol{x})) \\
&= -\mathrm{Cov}(\rho, D)\mathrm{Var}(D)^{-1}\mathrm{Cov}(D, w(\boldsymbol{x})),
\end{aligned}$$

recalling that $\mathrm{Cov}(\rho, w(\boldsymbol{x}))$ was assumed to be zero.

For new $\boldsymbol{x}$ for which the fast simulator has not been evaluated, the equations remain almost identical, but there is the added complication that $u'(\boldsymbol{x})$, the residual in the fast emulator, is not known. Since this appears in the final element of $b(\boldsymbol{x})$, the above equations cannot be evaluated. Under the assumption that the slow simulator runs D provide no further information about the fast simulator, the final element of $b(\boldsymbol{x})$ in these equations can be treated as fixed at the adjusted expectation $E_{\mathrm{D}'}(u'(\boldsymbol{x}))$.

A3 Building the emulators

A3.1 Choosing basis functions and removing inactive parameters

From experience, polynomial terms are often suitable choices. For each of the 75 outputs, linear models were built with (i) first-order (linear) terms only; (ii) second-order (quadratic) and first-order terms, with interactions; (iii) third-order (cubic) and lower-order terms, with first-order interactions. Explicitly, these are the models

$$f'(\boldsymbol{x}) = \sum_i a_i x_i + u'(\boldsymbol{x})$$

$$f'(\boldsymbol{x}) = \sum_i a_i x_i^2 + \sum_i \sum_{j \neq i} b_{ij} x_i x_j + \sum_i c_i x_i + u'(\boldsymbol{x})$$

$$f'(\boldsymbol{x}) = \sum_i a_i x_i^3 + \sum_i b_i x_i^2 + \sum_i \sum_{j \neq i} c_{ij} x_i x_j + \sum_i d_i x_i + u'(\boldsymbol{x}),$$

where the a_i, b_i, c_i, d_i collectively form the β_i' in Eq. (11) (and are, of course, different values in the three different models). Note that "linear" in "linear model" refers to the linearity of the form $\sum_i \beta_i g_i(\boldsymbol{x})$, not the linearity of the g_i, so all three models here are linear models.

The adjusted R^2 was examined for each model. The findings of this procedure, when applied to the fast simulator runs, can be summarised as follows.

- The models with only first-order terms were inadequate in many cases, leading to low R^2 and high residual variance. For some of the outputs they did provide good fits (adjusted R^2 between 0.9 and 0.95).

- The second-order models were very good (R^2 over 0.95) for almost every region and good for all regions (with the lowest R^2 of 0.89).

- The third-order models provide no noticeable improvements over second-order models.

As a result of this, the chosen g_i were second-order and lower terms for all outputs (that is, for all regions).

The second stage of emulation is the removal of inactive parameters. In the linear model for any given output quantity, most of the parameters have little impact. Emulators can be improved by focusing on a few important parameters and leaving the rest out of the mean trend entirely. This involves adding a small "nugget" of variance into the emulator, uncorrelated with everything else. This nugget represents the fact that now the emulator does not exactly predict the simulator output even at parameters already sampled, because some parameters have been ignored. For example, if only parameters x_1 and x_2 are active, then the emulator will give the same prediction whatever the value of $x_3, \ldots$, even though the simulator will give slightly different output in each case. The nugget accounts for this uncertainty. An estimate for the size of the nugget was derived by running the simulator with different values of the inactive parameters and observing the impact. This is a rather crude approach, but since the observed variation was several orders of magnitude lower than the other variances in the emulator, there is little benefit to a more careful analysis. Formally, the emulator becomes

$$f(\boldsymbol{x}) = \sum \beta_i g_i(\boldsymbol{x}_A) + u(\boldsymbol{x}_A) + v(\boldsymbol{x}),$$

where $\boldsymbol{x}_A$ are the active parameters, and $v(\boldsymbol{x})$ represents the nugget, with expectation zero, low variance and zero correlation with everything else.

A policy of stepwise elimination was followed for each output: at each step, each parameter was removed in turn, and the change in R^2 was calculated. The parameter whose removal caused the smallest change in this was removed. This process was continued for each output until either four parameters were left or the removal of a single parameter would reduce the R^2 by more than 0.03. A third criterion, that the R^2 should not be allowed to fall below some critical value, was considered but turned out to be unnecessary. The choice to require at least four parameters was made after observing that emulators with fewer parameters tended to perform poorly in validation.

In a standard emulation this would conclude the removal of inactive parameters, but since in this case the fast emulator is to be linked to the slow emulator, it is important to check that there are no parameters being removed that are much more important for the slow emulator. For this reason, the same stepwise selection was performed using the 200 runs of the slow simulator (ignoring the link with the fast emulator). This procedure selected the same parameters in most cases, occasionally with one difference. It is likely this is caused by

small quasi-random differences in the R^2, but for safety these parameters were also added back into the emulators. This led to an extra parameter being activated for four of the outputs.

Finally, since parameters x_4 and x_5 were closely related (the parameters governing the gamma distribution from which the PSD was calculated), it was decided that an active x_5 should lead to an active x_4 as well.

A3.2 Emulating the fast simulator

Since 1700 is a large number of runs, it is reasonable to make the simplification that the least-square estimates for β_i' are known quantities and that the residual variance can be used for $\mathrm{Var}(u'(\boldsymbol{x}))$. The only remaining task for the fast simulator's emulator is to specify the correlation. A squared correlation is used, that is,

$$\mathrm{Corr}\left(u'(\boldsymbol{x}_1), u'(\boldsymbol{x}_2)\right) = \exp\left(-\left(\frac{d(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\delta'}\right)^2\right),$$

where δ', the correlation length, is to be set, and $d(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the distance between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. In Appendix A1, some possibilities for choosing δ' are provided. Note that using a different scaling parameter for each dimension of the parameter space can be necessary in many cases, but for this application a single value proved sufficient (recall that all parameters have been normalised so they are all in [0, 1], otherwise different δ' would be needed for each dimension). The approach used in this application is to begin with $\delta' = 1/3$, then use leave-one-out validation using $f'(\mathcal{X}_F)$ to tune δ'. This tuning was done by varying δ' and observing the impact on the adjusted variance and the proportion of observations correctly predicted to within 3 standard deviations. Good choices of δ' lead to low adjusted variances but correctly predict most of the observations (at least 95 %). There is some judgement required on the balance between these competing criteria. A final check is to predict $f'(\mathcal{X}_S)$ using $f'(\mathcal{X}_F)$ and the chosen δ' to check that the method has been successful (again, by checking the proportion of observations that were predicted to within 3 standard deviations). This strategy suggested rather small values for the correlations lengths in all the regions, between 0.1 and 0.15.

Predictions of the remaining 200 runs using the emulator built from the first 1500 were accurate for all the outputs: an example can be seen in Fig. 5, for the case of the first output in the first hour. The emulator predictions are close to the observed output (that is, $f'(\mathcal{X}_S)$) relative to the emulator variances in most cases, and the emulator variances are small relative to the overall variability of simulator output across the parameter space. This analysis suggests that the emulator and the choices of δ' are appropriate. A final fast emulator was then built using all the runs ($f'(\mathcal{X}_S \cup \mathcal{X}_F)$) and the values of δ' calculated by the above method.

A4 Overview of process for multi-level emulation

This section contains the list of steps performed in the multi-level emulation approach used in this application and the appropriate reference within the paper for details.

- Generate a large number of fast simulator evaluations, with parameters chosen by an appropriate space-filling design such as Latin hypercube sampling (Sect. 4).

- Generate a smaller number of slow simulator evaluations. The fast simulator should be evaluated at these parameters as well (Sect. 4).

- Choose basis functions g_i for the emulators, through exploration and examination of regression fits (Sect. 6.1 and Appendix A3.1).

- Determine and remove inactive parameters (Sect. 6.1 and Appendix A3.1).

- Estimate the "nugget" created by the removal of inactive parameters (Appendix A3.1).

- Use least-square regression to estimate β_i' and the variance of u' (Sects. 5.2, 6.2 and Appendix A1).

- Choose, through judgements and exploration, a suitable correlation function to use for u' (Appendix A1 and A3.2).

- Using the fast simulator runs, the estimates from the least-square regression, and the chosen correlation function, construct a Bayes linear adjustment for the fast emulator (Sect. 5.2).

- Use diagnostic techniques to assess the validity of the fast emulator and tune parameters in the correlation function (Sect. 5.4, Appendix A1 and A3.2). If validation fails, return to a previous step and try something else.

- Link the fast and slow emulators (Sect. 5.3) and compute prior quantities for the Bayes linear adjustment (Sect. 5.3 and Appendix A1.1).

- Using this link, the fast emulator, and the slow simulator evaluations, perform a Bayes linear adjustment for the slow emulator (Sect. 6.3 and Appendix A2.

- Use diagnostic techniques to assess the validity of the slow emulator and tune parameters in the correlation function (Sects. 5.4, Sect. 6.3 and Appendix A3.2). If validation fails, return to a previous step and try something else.

References

Arason, P., Petersen, G., and Bjornsson, H.: Observations of the altitude of the volcanic plume during the eruption of Eyjafjallajökull, April–May 2010, Earth Syst. Sci. Data, 3, 9–17, https://doi.org/10.5194/essd-3-9-2011, 2011.

Bonadonna, C. and Phillips, J. C.: Sedimentation from strong volcanic plumes, J. Geophys. Res.-Solid, 108, 2340–2368, https://doi.org/10.1029/2002JB002034, 2003.

Casadevall, T. J.: The 1989–1990 eruption of Redoubt Volcano, Alaska: impacts on aircraft operations, J. Volcanol. Geoth. Res., 62, 301–316, 1994.

Costa, A., Macedonio, G., and Folch, A.: A three-dimensional Eulerian model for transport and deposition of volcanic ashes, Earth Planet. Sc. Lett., 241, 634–647, 2006.

Cressie, N.: Statistics for spatial data, Wiley, USA, 1993.

Cumming, J. and Goldstein, M.: Small sample bayesian designs for complex high-dimensional models based on information gained using fast approximations, Technometrics, 51, 377–388, 2009.

Dacre, H., Grant, A., Harvey, N., Thomson, D., Webster, H., and Marenco, F.: Volcanic ash layer depth: Processes and mechanisms, Geophys. Res. Lett., 42, 637–645, 2015.

Dacre, H. F., Grant, A. L. M., Hogan, R. J., Belcher, S. E., Thomson, D. J., Devenish, B., Marenco, F., Haywood, J. M., Ansmann, A., Mattis, I., and Clarisse, L.: Evaluating the structure and magnitude of the ash plume during the initial phase of the 2010 Eyjafjallajökull eruption using lidar observations and NAME simulations, J. Geophys. Res., 116, D00U03, https://doi.org/10.1029/2011JD015608, 2011.

Dacre, H. F., Grant, A. L. M., and Johnson, B. T.: Aircraft observations and model simulations of concentration and particle size distribution in the Eyjafjallajökull volcanic ash cloud, Atmos. Chem. Phys., 13, 1277–1291, https://doi.org/10.5194/acp-13-1277-2013, 2013.

Devenish, B., Thomson, D., Marenco, F., Leadbetter, S., Ricketts, H., and Dacre, H.: A study of the arrival over the United Kingdom in April 2010 of the Eyjafjallajökull ash cloud using ground-based lidar and numerical simulations, Atmos. Environ., 48, 152–164, 2012a.

Devenish, B. J., Francis, P. N., Johnson, B. T., Sparks, R. S. J., and Thomson, D. J.: Sensitivity analysis of dispersion modeling of volcanic ash from Eyjafjallajökull in May 2010, J. Geophys. Res., 117, D13208, https://doi.org/10.1029/2011JD016782, 2012b.

Dürbeck, T. and Gerz, T.: Large-eddy simulation of aircraft exhaust plumes in the free atmosphere: Effective diffusivities and cross-sections, Geophys. Res. Lett., 22, 3203–3206, 1995.

European Commission: Air passenger transport by reportingcountry, Eurostat, Subset used: 2015, European Union (28 countries), available at: http://ec.europa.eu/eurostat/web/products-datasets/-/avia_paoc (last access: 15 March 2017), 2016.

Folch, A., Costa, A., and Basart, S.: Validation of the FALL3D ash dispersion model using observations of the 2010 Eyjafjallajökull volcanic ash clouds, Atmos. Environ., 48, 165–183, 2012.

Francis, P. N., Cooke, M. C., and Saunders, R. W.: Retrieval of physical properties of volcanic ash using Meteosat: A case study from the 2010 Eyjafjallajökull eruption, J. Geophys. Res., 117, D00U09, https://doi.org/10.1029/2011JD016788, 2012.

Girard, S., Korsakissok, I., and Mallet, V.: Screening sensitivity analysis of a radionuclides atmospheric dispersion model applied to the Fukushima disaster, Atmos. Environ., 95, 490–500, 2014.

Goldstein, M. and Huntley, N.: Bayes linear emulation, history matching and forecasting for complex computer simulators, in: Handbook of Uncertainty Quantification, Springer, Cham, Switzerland, 2017.

Goldstein, M. and Wooff, D.: Bayes linear statistics, theory and methods, John Wiley & Sons, UK, 2007.

Goldstein, M., Seheult, A., and Vernon, I.: Assessing Model Adequacy, Environmental Modelling: Finding Simplicity in Complexity, 2nd Edn., John Wiley & Sons, Ltd, Chichester, UK, 435–449, 2010.

Grant, A. L. M., Dacre, H. F., Thomson, D. J., and Marenco, F.: Horizontal and vertical structure of the Eyjafjallajökull ash cloud over the UK: a comparison of airborne lidar observations and simulations, Atmos. Chem. Phys., 12, 10145–10159, https://doi.org/10.5194/acp-12-10145-2012, 2012.

Harris, G. R., Collins, M., Sexton, D. M. H., Murphy, J. M., and Booth, B. B. B.: Probabilistic projections for 21st century European climate, Nat. Hazards Earth Syst. Sci., 10, 2009–2020, https://doi.org/10.5194/nhess-10-2009-2010, 2010.

Hobbs, P. V., Radke, L. F., Lyons, J. H., Ferek, R. J., Coffman, D. J., and Casadevall, T. J.: Airborne measurements of particle and gas emissions from the 1990 volcanic eruptions of Mount Redoubt, J. Geophys. Res., 96, 18735–18752, 1991.

Johnson, B., Turnbull, K., Brown, P., Burgess, R., Dorsey, J., Baran, A. J., Webster, H., Haywood, J., Cotton, R., Ulanowski, Z., Hesse, E., Woolley, A., and Rosenberg, P.: In situ observations of volcanic ash clouds from the FAAM aircraft during the eruption of Eyjafjallajökull in 2010, J. Geophys. Res., 117, D00U24, https://doi.org/10.1029/2011JD016760, 2012.

Johnson, J., Cui, Z., Lee, L., Gosling, J., Blyth, A., and Carslaw, K.: Evaluating uncertainty in convective cloud microphysics using statistical emulation, J. Adv. Model. Earth Syst., 7, 162–187, 2015.

Jones, A., Thomson, D., Hort, M., and Devenish, B.: The UK Met Office's next-generation atmospheric dispersion model,

NAME III, in: Air Pollution Modeling and its Application XVII, Springer, New York, 580–589, 2007.

Lee, L. A., Carslaw, K. S., Pringle, K. J., Mann, G. W., and Spracklen, D. V.: Emulation of a complex global aerosol model to quantify sensitivity to uncertain parameters, Atmos. Chem. Phys., 11, 12253–12273, https://doi.org/10.5194/acp-11-12253-2011, 2011.

Lee, L. A., Carslaw, K. S., Pringle, K. J., and Mann, G. W.: Mapping the uncertainty in global CCN using emulation, Atmos. Chem. Phys., 12, 9739–9751, https://doi.org/10.5194/acp-12-9739-2012, 2012.

Lee, L. A., Pringle, K. J., Reddington, C. L., Mann, G. W., Stier, P., Spracklen, D. V., Pierce, J. R., and Carslaw, K. S.: The magnitude and causes of uncertainty in global model simulations of cloud condensation nuclei, Atmos. Chem. Phys., 13, 8879–8914, https://doi.org/10.5194/acp-13-8879-2013, 2013.

Marenco, F., Johnson, B., Turnbull, K., Newman, S., Haywood, J., Webster, H., and Ricketts, H.: Airborne lidar observations of the 2010 Eyjafjallajökull volcanic ash plume, J. Geophys. Res., 116, D00U05, https://doi.org/10.1029/2011JD016396, 2011.

Maryon, R. H.: The gravitational settling of particulates: towards a parameterisation for the NAME dispersion model, Met O (APR) Turbulence and Diffusion Note No. 244, UK Met Office, Bracknell, 1997.

Maryon, R. H., Ryall, D. B., and Malcolm, A. L.: The NAME dispersion model: Science documentation, MO (PMSR) Turbulence and Diffusion Note No. 262, UK Met Office, Bracknell, 1999.

Mastin, L., Guffanti, M., Servranckx, R., Webley, P., Barsotti, S., Dean, K., Durant, A., Ewert, J., Neri, A., Rose, W., Schneider, D., Siebert, L., Stunder, B., Swanson, G., Tupper, A., Volentik, A., and Waythomas, C.: A multidisciplinary effort to assign realistic source parameters to models of volcanic ash-cloud transport and dispersion during eruptions, J. Volcanol. Geoth. Res., 186, 10–21, 2009.

Mazzocchi, M., Hansstein, F., and Ragona, M.: The volcanic ash cloud and its financial impact on the European airline industry, CESifo Forum No. 2, 92–100, 2010.

Mulder, K. J., Lickiss, M., Harvey, N., Black, A., Charlton-Perez, A., Dacre, H., and McCloy, R.: Visualizing Volcanic Ash Forecasts: Scientist and Stakeholder Decisions Using Different Graphical Representations and Conflicting Forecasts, Weather Clim. Soc., 9, 333–348, 2017.

Pappalardo, G.: Four-dimensional distribution of the 2010 Eyjafjallajökull volcanic cloud over Europe observed by EARLINET, Atmos. Chem. Phys., 13, 4429–4450, https://doi.org/10.5194/acp-13-4429-2013, 2013.

Saltelli, A. and Annoni, P.: How to avoid a perfunctory sensitivity analysis, Environmental Model. Softw., 25, 1508–1517, 2010.

Sarri, A., Guillas, S., and Dias, F.: Statistical emulation of a tsunami model for sensitivity analysis and uncertainty quantification, Nat. Hazards Earth Syst. Sci., 12, 2003–2018, https://doi.org/10.5194/nhess-12-2003-2012, 2012.

Schumann, U., Konopka, P., Baumann, R., Busen, R., Gerz, T., Schalger, H., Schulte, P., and Volkert, H.: Estimate of diffusion parameters f aircraft exhaust plumes near the tropopause from nitric oxide and turbulence measurements, J. Geophys. Res., 100, 14147–14162, 1995.

Schumann, U., Weinzierl, B., Reitebuch, O., Schlager, H., Minikin, A., Forster, C., Baumann, R., Sailer, T., Graf, K., Mannstein,

H., Voigt, C., Rahm, S., Simmet, R., Scheibe, M., Lichtenstern, M., Stock, P., Rüba, H., Schäuble, D., Tafferner, A., Rautenhaus, M., Gerz, T., Ziereis, H., Krautstrunk, M., Mallaun, C., Gayet, J.-F., Lieke, K., Kandler, K., Ebert, M., Weinbruch, S., Stohl, A., Gasteiger, J., Groß, S., Freudenthaler, V., Wiegner, M., Ansmann, A., Tesche, M., Olafsson, H., and Sturm, K.: Airborne observations of the Eyjafjallajökull volcano ash cloud over Europe during air space closure in April and May 2010, Atmos. Chem. Phys., 11, 2245–2279, https://doi.org/10.5194/acp-11-2245-2011, 2011.

Sparks, R. S. J., Bursik, M., Carey, S., Gilbert, J., Glaze, L., Sigurdsson, H., and Woods, A.: Volcanic plumes, Wiley, Chichester, 1997.

Underwood, B.: Review of deposition velocity and washout coefficient, http://www.hpa.org.uk/webc/HPAwebFile/HPAweb_C/1194947314056 (last access: February 2016), 2011.

Urban, N. M. and Fricker, T. E.: A comparison of Latin hypercube and grid ensemble designs for the multivariate emulation of an Earth system model, Comput. Geosci., 36, 746–755, 2010.

Vernon, I., Goldstein, M., and Bower, R. G.: Galaxy formation: a Bayesian uncertainty analysis, Bayesian Analysis, 5, 619–669, 2010.

Webley, P., Stunder, B., and Dean, K.: Preliminary sensitivity study of eruption source parameters for operational volcanic ash cloud transport and dispersion models – A case study of the August 1992 eruption of the Crater Peak vent, Mount Spurr, Alaska, J. Volcanol. Geoth. Res., 186, 108–119, 2009.

Webster, H. and Thomson, D.: Dry deposition modelling in a Lagrangian dispersion model, Int. J. Environ. Pollut., 47, 1–9, 2011.

Webster, H. N., Thomson, D. J., Johnson, B. T., Heard, I. P. C., Turnbull, K., Marenco, F., Kristiansen, N. I., Dorsey, J., Minikin, A., Weinzierl, B., Schumann, U., Sparks, R. S. J., Loughlin, S. C., Hort, M. C., Leadbetter, S. J., Devenish, B. J., Manning, A. J., Witham, C. S., Haywood, J. M., and Golding, B. W.: Operational prediction of ash concentrations in the distal volcanic cloud from the 2010 Eyjafjallajökull eruption, J. Geophys. Res., 117, D00U08, https://doi.org/10.1029/2011JD016790, 2012.

Webster, H. N. and Thomson, D. J.: Parameterising low-frequency meander in atmospheric dispersion models, in: Proceedings of the 10th Int. Conf. on Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes, 17–20 October 2005, Sissi, Crete, 594–598, 2005.

Webster, H. N. and Thomson, D. J.: The NAME wet deposition scheme, Met Office Forecasting Research Technical Report No. 584, Met Office, Exeter, UK, 2014.

Webster, H. N., Whitehead, T., and Thomson, D. J.: Parametrizing unresolved mesoscale motions, Met Office Forecasting Research Technical Report No. 601, Met Office, Exeter, UK, 2015.

Witham, C., Hort, M., Potts, R., Servranckx, R., Husson, P., and Bonnardot, F.: Comparison of VAAC atmospheric dispersion models using the 1 November 2004 Grimsvötn eruption, Meteorol. Appl., 14, 27–38, 2007.

Witham, C., Hort, M., Thomson, D., Leadbetter, S., Devenish, B., and Webster, H.: The current volcanic ash modelling setup at the London VAAC, http://www.metoffice.gov.uk/binaries/content/assets/mohippo/pdf/aviation/vaac/london_vaac_current_modelling_setup_v01-3_12012017.pdf (last access: 7 September 2017), 2012a.

Witham, C., Webster, H., Hort, M., Jones, A., and Thomson, D.: Modelling concentrations of volcanic ash encountered by aircraft in past eruptions, Atmos. Environ., 48, 219–229, https://doi.org/10.1016/j.atmosenv.2011.06.073, 2012b.

Woodhouse, M. J., Hogg, A. J., Phillips, J. C., and Sparks, R. S. J.: Interaction between volcanic plumes and wind during the 2010 Eyjafjallajökull eruption, Iceland, J. Geophys. Res., 118, 92–109, https://doi.org/10.1029/2012JB009592, 2013.