

Simultaneous model calibration and source inversion in atmospheric dispersion models

Juan G. García · Bamdad Hosseini · John M. Stockie

June 14, 2018

Abstract We present a cost-effective method for model calibration and solution of source inversion problems in atmospheric dispersion modelling. We use Gaussian process emulations of atmospheric dispersion models within a Bayesian framework for solution of inverse problems. The model and source parameters are treated as unknowns and we obtain point estimates and approximation of uncertainties for sources while simultaneously calibrating the forward model. The method is validated in the context of an industrial case study involving emissions from a smelting operation for which cumulative monthly measurements of zinc particulate depositions are available.

Keywords Atmospheric dispersion · Particle deposition · Inverse source identification · Bayesian estimation · Uncertainty quantification

Mathematics Subject Classification (2000) 62F15 · 65M08 · 65M32 · 86A10

1 Introduction

As a result of growing concerns over anthropogenic impacts on the environment and the resulting climate changes, studies of atmospheric dispersion models have proliferated over recent years. Mathematical models for atmospheric pollutant dispersion provide approximations to the actual physical processes driving dispersion of particulates and other substances within the atmosphere [2, 20]. A common challenge in dispersion modelling is to maintain accuracy since many models suffer from sizable errors and large uncertainties in parameters. In some cases, these inaccuracies can be addressed by increasing the model complexity to account for physical processes that have been neglected, in the hope that the resulting more complex model yields correspondingly higher accuracy and more realistic simulations. However, in many practical applications, such extensions lead to no significant improvement in the solution because the model errors are far outweighed by the uncertainty in values of the model parameters. This difficulty is further exacerbated by the fact that many models contain a large number of unknown model parameters. Some of these parameters have a clear physical basis, such as diffusion coefficients, particulate density and settling velocities, whereas others are mathematical (or fitting) parameters such as the Monin-Obukhov length, atmospheric stability classes, or terrain roughness length.

Direct measurements of many physical and mathematical parameters are often problematic or even infeasible; for example, eddy diffusion coefficients are typically difficult to estimate in practice [34]. One is therefore often forced to deal with significant uncertainties in the true value of model parameters, even when complex models are employed that under ideal circumstances could potentially simulate results very accurately. Because large errors in model parameters can severely contaminate simulations, it is therefore essential when simulating atmospheric dispersion scenarios to simultaneously understand and control the

J.G. García
3DM Devices Inc. 26019 31B, Aldergrove, BC V4W 2Z6, Canada
E-mail: juan_g@3dm.com

J.M. Stockie
Department of Mathematics, Simon Fraser University, 8888 University Drive, Burnaby, BC V5A 1S6, Canada
E-mail: jstockie@sfu.ca

B. Hosseini
Computing and Mathematical Sciences, California Institute of Technology, 1200 E. California Blvd., Pasadena, CA 91125, USA
E-mail: bamdadh@caltech.edu

parameter uncertainty. Sources of uncertainty are diverse and can be classified roughly into one of three types [29]:

- Data uncertainty: when the empirical measurements or model parameter values are inaccurate;
- Model uncertainty: where the model does not capture all physical processes of interest; or
- Stochastic uncertainty: relating to the inherent unpredictability or randomness in physical processes, such as atmospheric turbulence.

In this work we focus primarily on dealing with the first case of data uncertainty appearing in the context of atmospheric source inversion problems.

Despite the difficulty of obtaining accurate simulations, there is a growing need to develop more accurate and computationally efficient models for practical atmospheric dispersion applications [19]. The necessity for computational efficiency comes from the fact that models for atmospheric transport are usually expressed as systems of partial differential equations that are computationally expensive to solve at the typical resolutions required, which is a particular limitation for real-time applications. Efficiency is of even greater importance in the context of source inversion problems, since a single solution of the inverse problem may require a large number of evaluations of the forward dispersion model. One of the goals of this work is therefore to demonstrate how to use computationally demanding forward models to solve atmospheric source inversion problems in an efficient manner.

1.1 Atmospheric dispersion modelling

We model the dispersion of pollutants in the atmosphere using the advection-diffusion partial differential equation (PDE)

$$\frac{\partial C(\mathbf{x}, t)}{\partial t} + \nabla \cdot (\bar{\mathbf{v}}C(\mathbf{x}, t) + \mathbf{D}\nabla C(\mathbf{x}, t)) = f(\mathbf{x}, t), \quad (1)$$

where $C(\mathbf{x}, t)$ denotes the concentration of pollutant at location $\mathbf{x} = (x, y, z) \in \mathbb{R}^3$, t is time, $\bar{\mathbf{v}}$ is the wind velocity field, \mathbf{D} is the diffusivity tensor, and $f(\mathbf{x}, t)$ is a pollutant source term. We focus on the case of a finite collection of n point sources each having an emission rate that is constant in time, which allows f to be written as

$$f(\mathbf{x}, t) = \sum_{j=1}^n q_j \delta(\mathbf{x} - \mathbf{x}_j), \quad (2)$$

where $\mathbf{q} = (q_1, q_2, \dots, q_n)^T \in \mathbb{R}^n$ represent emission rates and $\delta(\mathbf{x} - \mathbf{x}_j)$ is the Dirac delta function for the point source centered at location \mathbf{x}_j . The wind velocity field $\bar{\mathbf{v}}$ and diffusivity tensor \mathbf{D} are often difficult to measure and so are replaced using simpler mathematical approximations that depend upon empirical parameters describing the variation of $\bar{\mathbf{v}}$ and \mathbf{D} with time and space. Common examples of such parameters include roughness length, Monin-Obukhov length and Pasquill stability class [34]. We combine all empirical parameters together into a vector $\boldsymbol{\theta} = (\theta_k) \in \mathbb{R}^m$, and then denote the linear operator $\nabla \cdot (\mathbf{v} + \mathbf{D}\nabla)$ on the left hand side of (1) by $\mathbf{L}(\boldsymbol{\theta})$ to make explicit the dependence of the operator on the parameter vector $\boldsymbol{\theta}$, noting that the dependence of \mathbf{L} on $\boldsymbol{\theta}$ can still be nonlinear. Then we rewrite (1) as

$$\frac{\partial C(\mathbf{x}, t)}{\partial t} + \mathbf{L}(\boldsymbol{\theta})C(\mathbf{x}, t) = \sum_{j=1}^n q_j \delta(\mathbf{x} - \mathbf{x}_j). \quad (3)$$

The task of finding the concentration $C(\mathbf{x}, t)$ given \mathbf{q} and $\boldsymbol{\theta}$, supplemented by suitable boundary and initial conditions, is referred to as the *forward problem* in the context of atmospheric pollutant transport.

1.2 Atmospheric source inversion

The goal of atmospheric source inversion is to estimate the source term $f(\mathbf{x}, t)$ on the right hand side of (1) from indirect measurements of the concentration $C(\mathbf{x}, t)$. We consider equation (3) with a set of given point sources at known locations, but with unknown constant emission rates. Our aim is then to estimate the emission rates q_j from indirect measurements of $C(\mathbf{x}, t)$. To this end, the source inversion problem is the reverse of the forward problem (3) and is thus an *inverse problem*.

We assume that measurements of concentration can be treated in terms of the action of bounded linear operators on C . For example, a common measurement in particulate dispersion studies is the total

accumulated deposition of pollutants in a region $R \subset \mathbb{R}^2$ of the ground surface after some time T has elapsed. Such a measurement can be written as

$$\int_R \int_0^T C(x, y, 0, t) v_{set} dt dx dy,$$

where v_{set} is the vertical settling velocity at the ground surface for the pollutant of interest. The class of all possible linear measurements is large and includes many commonly used methods in practice ranging from short time average measurements such as Xact ambient metal monitors [4] and Andersen high-volume air samplers [40], as well as averaged long-time measurements accumulated in dustfall jars.

Suppose now that field measurements of C are taken at d locations in space and collected in a vector $\mathbf{w} \in \mathbb{R}^d$. Since the measurements are linear, we can write

$$\mathbf{w} = \mathcal{A}(\boldsymbol{\theta}) \mathbf{q}, \quad (4)$$

where $\mathcal{A}(\boldsymbol{\theta})$ is a $d \times n$ matrix that represents the solution map of the PDE (3) and depends nonlinearly on the model parameters $\boldsymbol{\theta}$. Source inversion requires finding \mathbf{q} given \mathbf{w} , and it is well known that this problem is ill-posed in the sense of Hadamard [6, 13]; that is, small variations in the inputs (\mathbf{w}) result in large variations in the solution (\mathbf{q}). This is the case even if the true value of $\boldsymbol{\theta}$ is known exactly.

A variety of different approaches have been proposed in the literature to solve source inversion problems [10, 30]. Lin and Chang [21] used an air trajectory statistical approach to estimate the strength of different sources of volatile organic compounds of anthropogenic origin. Stockie and Lushi [22] used a Gaussian plume approximation to the governing PDEs to estimate ground-level deposition of zinc from a lead-zinc smelter, using linear least-squares to perform the source inversion. Skiba [35] solved the adjoint equation for the advection-diffusion equation and used penalized least-squares to invert the sources.

All of the methods just mentioned yield point estimates for the source strengths but provide no direct measure of uncertainty in the estimated parameters. This drawback is overcome by the use of probabilistic methods such as the Bayesian approach for solving inverse problems. Such methods provide a robust setting for solving the inverse problem and quantifying the uncertainties associated with the solution. Sohn et al. [37] developed an algorithm to obtain estimates and uncertainties for the location and strength of pollutant sources in buildings using data obtained from a COMIS simulation. Hosseini and Stockie [11] used a Gaussian plume model combined with experimental data to estimate and quantify the uncertainty of airborne fugitive emissions. In a follow-up publication [12], the same authors coupled a Bayesian approach with a finite volume solver to estimate and quantify the strength of airborne contaminants from a fixed number of point sources at known locations. Keats et al. [17] obtained probability distributions for source strengths and locations using data from the standard ‘‘Muck Urban Setting Test’’ experiment.

The central contribution of this work is in proposing a method that deals with the calibration parameters $\boldsymbol{\theta}$ as unknowns. We infer $\boldsymbol{\theta}$ along with the vector of source strengths \mathbf{q} , which results in a nonlinear inverse problem that is solved within the Bayesian framework. The problem of estimating the true value of the parameters $\boldsymbol{\theta}$ is referred to as *model calibration*. Traditionally, the process of tuning parameters in atmospheric dispersion models is done empirically using Pasquill stability classes [34, 41] that are often chosen heuristically. Instead, we automatically estimate these parameters using the information contained in the measured data \mathbf{w} .

The remainder of this paper is organized as follows. In Section 2 we discuss the use of Gaussian process regression to emulate computationally expensive models. In Section 3 we introduce our Bayesian framework for simultaneous calibration of the model and solution to the source inversion problem. In Section 4 we apply the results of the previous sections to an industrial case study involving air-borne particulate emissions from a lead-zinc smelter in Trail, British Columbia, Canada.

2 Emulation of atmospheric dispersion models

Equation (3) captures the linear relationship between source strengths $\mathbf{q} \in \mathbb{R}^n$ and concentration $C(\mathbf{x}, t)$. Under the further assumption that the data $\mathbf{w} \in \mathbb{R}^d$ also depends linearly on concentration $C(\mathbf{x}, t)$, we can write the map $\mathbf{q} \mapsto \mathbf{w}$ in the linear form (4), although we note that in general, $\mathcal{A}(\boldsymbol{\theta}) \in \mathbb{R}^{d \times n}$ may depend nonlinearly on $\boldsymbol{\theta} \in \mathbb{R}^m$. Therefore, the map $(\mathbf{q}, \boldsymbol{\theta}) \mapsto \mathbf{w}$ can be split into two parts: for fixed values of $\boldsymbol{\theta}$ the map $(\cdot, \boldsymbol{\theta}) \mapsto \mathbf{w}$ is linear, whereas for a fixed \mathbf{q} the mapping $(\mathbf{q}, \cdot) \mapsto \mathbf{w}$ is nonlinear. We will exploit this structure in the design of our source inversion algorithm, using the fact that the linear map $(\cdot, \boldsymbol{\theta}) \mapsto \mathbf{q}$ can be dealt with efficiently, but that we will need to approximate the nonlinear map $(\mathbf{q}, \cdot) \mapsto \mathbf{w}$.

To this end, we approximate the matrix

$$\mathcal{A}(\boldsymbol{\theta}) = [\mathbf{a}_{ij}(\boldsymbol{\theta})] \in \mathbb{R}^{d \times n}, \quad (5)$$

with another matrix

$$A(\boldsymbol{\theta}) = [a_{ij}(\boldsymbol{\theta})] \in \mathbb{R}^{d \times n},$$

where each entry $\mathbf{a}_{ij} : \mathbb{R}^m \mapsto \mathbb{R}$ of \mathcal{A} is approximated by the map $a_{ij} : \mathbb{R}^m \mapsto \mathbb{R}$. For this purpose, we make use of the Gaussian process (GP) emulators of [18]. Emulation with GPs is a well-established method in dynamic computer experiments and machine learning and we refer the reader to [3, 18, 26, 31] and the references within for an introduction to this subject. Here, we outline our method for approximating \mathcal{A} and will not discuss the theory of GP emulators with the exception of a few crucial definitions and results.

Definition 1 A Gaussian process (GP) on \mathbb{R}^m is a collection of real-valued random variables $\{g(\mathbf{x})\}_{\mathbf{x} \in \mathbb{R}^m}$, any finite collection of which have a joint Gaussian distribution.

We denote a GP using the notation

$$g \sim \mathcal{GP}(\bar{g}, \kappa),$$

where the mean of the GP is

$$\bar{g}(\boldsymbol{\theta}) = \mathbb{E} g(\boldsymbol{\theta}) \quad \forall \boldsymbol{\theta} \in \mathbb{R}^m,$$

and κ is a positive semi-definite kernel with

$$\kappa(\boldsymbol{\theta}, \boldsymbol{\theta}') = \mathbb{E} (g(\boldsymbol{\theta}) - \bar{g}(\boldsymbol{\theta}))(g(\boldsymbol{\theta}') - \bar{g}(\boldsymbol{\theta}')) \quad \forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in \mathbb{R}^m.$$

We are primarily interested in isotropic kernels that satisfy

$$\kappa(\boldsymbol{\theta}, \boldsymbol{\theta}') = \kappa(|\boldsymbol{\theta} - \boldsymbol{\theta}'|),$$

from which it follows that if \bar{g} is continuous and κ is continuous at zero, then the GP is also mean square continuous [31].

Let $\{\boldsymbol{\theta}_k\}_{k=1}^K$ be a collection of *design points* in the parameter space \mathbb{R}^m for some fixed $K > 0$, and let $\{\mathbf{e}_j\}_{j=1}^n$ be the unit coordinate basis vectors in \mathbb{R}^n . Then for each index pair (i, j) define a GP

$$g_{ij} \sim \mathcal{GP}(\bar{g}_{ij}, \kappa_{ij}),$$

subject to the constraint

$$g_{ij}(\boldsymbol{\theta}_k) = \mathbf{a}_{ij}(\boldsymbol{\theta}_k) = (\mathcal{A}(\boldsymbol{\theta}_k)\mathbf{e}_j)_i \quad \text{for } k = 1, \dots, K.$$

That is, each g_{ij} interpolates the \mathbf{a}_{ij} at points $\{\boldsymbol{\theta}_k\}_{k=1}^K$. We can now identify g_{ij} using well-known identities for conditioning of Gaussian random variables [31, Sec. 2.2]. Let $\Theta := (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K)^T$ and define the following matrices and vectors

$$\begin{aligned} G_{ij}(\Theta, \Theta) &\in \mathbb{R}^{K \times K}, & [G_{ij}(\Theta, \Theta)]_{k\ell} &:= \kappa_{ij}(\boldsymbol{\theta}_k, \boldsymbol{\theta}_\ell), \\ G_{ij}(\boldsymbol{\theta}, \Theta) &\in \mathbb{R}^{1 \times K}, & [G_{ij}(\boldsymbol{\theta}, \Theta)]_\ell &:= \kappa_{ij}(\boldsymbol{\theta}, \boldsymbol{\theta}_\ell), \\ G_{ij}(\Theta, \boldsymbol{\theta}) &\in \mathbb{R}^{K \times 1}, & [G_{ij}(\Theta, \boldsymbol{\theta})]_k &:= \kappa_{ij}(\boldsymbol{\theta}_k, \boldsymbol{\theta}), \\ \mathbf{g}_{ij} &\in \mathbb{R}^{K \times 1}, & [\mathbf{g}_{ij}]_k &= \mathbf{a}_{ij}(\boldsymbol{\theta}_k), \end{aligned}$$

for $k, \ell = 1, \dots, K$. Then we have

$$g_{ij}(\boldsymbol{\theta}) \sim \mathcal{GP}(\bar{g}_{ij}(\boldsymbol{\theta}), \sigma_{ij}(\boldsymbol{\theta})),$$

where

$$\begin{aligned} \bar{g}_{ij}(\boldsymbol{\theta}) &= G_{ij}(\boldsymbol{\theta}, \Theta) G_{ij}(\Theta, \Theta)^{-1} \mathbf{g}_{ij}, \\ \sigma_{ij}(\boldsymbol{\theta}) &= \kappa_{ij}(\boldsymbol{\theta}, \boldsymbol{\theta}) - G_{ij}(\boldsymbol{\theta}, \Theta) \left[G_{ij}(\Theta, \Theta)^{-1} \right] G_{ij}(\Theta, \boldsymbol{\theta}), \end{aligned}$$

for all $\boldsymbol{\theta} \in \mathbb{R}^m$. Since the mean \bar{g}_{ij} interpolates the data $\mathbf{a}_{ij}(\boldsymbol{\theta}_k)$, it is also a good candidate for approximating \mathbf{a}_{ij} , and so we take

$$a_{ij}(\boldsymbol{\theta}) = \bar{g}_{ij}(\boldsymbol{\theta}) \quad \forall \boldsymbol{\theta} \in \mathbb{R}^m.$$

The advantage of using this approach for interpolation is that it is possible to assess the uncertainty and quality of the emulator via the covariance operator $\sigma(\boldsymbol{\theta})$.

The variance at a point $\boldsymbol{\theta}_i$ is given by the term $\sigma_{ii}(\boldsymbol{\theta})$ and is strongly influenced by the spatial distribution of the points $\{\boldsymbol{\theta}_k\}_{k=1}^K$ [14]. Thus, in order to emulate $\mathcal{A}(\boldsymbol{\theta})$, it is crucial to choose the points $\boldsymbol{\theta}_k$ in such a way that the uncertainty of the emulator is minimized. A popular method for choosing the

design points is known as a space-filling design [14], instances of which include the maximum entropy design [33], uniform design [7], and Latin Hypercube design (LHD) [24]. Following Jones et al. [15] we use a combination of the LHD and maximin designs, which is introduced in [14] and shown to outperform most space-filling design methods for Gaussian process interpolation.

We now briefly outline our space-filling design procedure. The main idea in the LHD is to distribute design points such that low-dimensional projections of the points do not overlap. One issue with LHD is that it does not have desirable space-filling properties and so leads to large uncertainties in the emulator in high dimensions. Thus, LHD is often used as an initial condition for other space-filling design methodologies and so here we complement the LHD with an approximate maximin design [14].

The idea behind the maximin design is as follows. Let $T \subset \mathbb{R}^n$ be the subset of parameter space in which we wish to construct our design points and consider all subsets S of T with finite (fixed) cardinality, say $|S| = k$. A maximin design S^o is a set of points that satisfies

$$\max_{S \subset T, |S|=k} \min_{s, s' \in S} d(s, s') = \min_{s, s' \in S^o} d(s, s'), \quad (6)$$

where d is the Euclidean metric. The optimization problem in equation (6) is not easily solvable, which is why we resort to metaheuristic optimization procedures. In particular, we use the particle swarm algorithm [1], initiated with an LHD. We then apply a number of iterations of particle swarm that is chosen depending on our available computational budget. In Figure 3 we show an example of the final design after 10,000 steps of a particle swarm algorithm.

3 The Bayesian inversion framework

We now turn to the source inversion problem, where the Bayesian approach combines the data \mathbf{w} together with prior knowledge to give a posterior probability distribution on the source strengths \mathbf{q} and calibration parameters $\boldsymbol{\theta}$. Following Bayes' rule, the posterior distribution of $(\boldsymbol{\theta}, \mathbf{q})$ is given by

$$\mathbb{P}_{post}(\boldsymbol{\theta}, \mathbf{q} | \mathbf{w}) = \frac{\mathbb{P}_{like}(\mathbf{w} | \boldsymbol{\theta}, \mathbf{q}) \mathbb{P}_{prior}(\boldsymbol{\theta}, \mathbf{q})}{Z(\mathbf{w})}, \quad (7)$$

where $\mathbb{P}_{like}(\mathbf{w} | \boldsymbol{\theta}, \mathbf{q})$ is the *likelihood probability* of \mathbf{w} given both $\boldsymbol{\theta}$ and \mathbf{q} , which represents the probability of observing \mathbf{w} given fixed values \mathbf{q} and $\boldsymbol{\theta}$. The probability distribution $\mathbb{P}_{prior}(\boldsymbol{\theta}, \mathbf{q})$ is called the *prior probability* that expresses prior knowledge of the possible values of \mathbf{q} and $\boldsymbol{\theta}$, before observing any experimental data. Finally, $Z(\mathbf{w})$ is a constant that normalizes the posterior probability so that it integrates to one, leading to the requirement that

$$Z(\mathbf{w}) = \int \mathbb{P}_{like}(\mathbf{w} | \boldsymbol{\theta}, \mathbf{q}) \mathbb{P}_{prior}(\boldsymbol{\theta}, \mathbf{q}) d\boldsymbol{\theta} d\mathbf{q}. \quad (8)$$

We next assume that the linearity requirement (4) holds for physical measurements up to some additive Gaussian measurement noise, so that

$$\mathbf{w} = \mathcal{A}(\boldsymbol{\theta})\mathbf{q} + \epsilon \quad \text{with} \quad \epsilon \sim \mathcal{N}(0, \Sigma). \quad (9)$$

Here, ϵ is a normally distributed random vector with mean zero and covariance $\Sigma \in \mathbb{R}^{d \times d}$, which is assumed positive-definite. The measurement noise ϵ models the deviations in the measurements \mathbf{w} due to missing physics and uncertainty in measurements. Under this assumption, it can be readily shown that [16, Sec. 3.2.1]

$$\mathbb{P}_{like}(\mathbf{w} | \boldsymbol{\theta}, \mathbf{q}) = \frac{1}{(2\pi \det \Sigma)^{\frac{1}{2}}} \exp\left(-\frac{1}{2} \left\| \Sigma^{-1/2} (\mathcal{A}(\boldsymbol{\theta})\mathbf{q} - \mathbf{w}) \right\|^2\right). \quad (10)$$

The choice of the prior distribution depends on previous knowledge about the model parameters and so is problem specific. For atmospheric dispersion models, the parameter $\boldsymbol{\theta}$ often depends on atmospheric conditions, whereas the emission rates in the vector \mathbf{q} depend on the physical processes that generated the emissions. Consequently, it is reasonable to assume a priori that \mathbf{q} and $\boldsymbol{\theta}$ are statistically independent, so that

$$\mathbb{P}_{prior}(\boldsymbol{\theta}, \mathbf{q}) = \mathbb{P}_{prior}(\boldsymbol{\theta}) \mathbb{P}_{prior}(\mathbf{q}). \quad (11)$$

In most cases, one chooses a prior for \mathbf{q} that imposes a positivity constraint on the emission rates, whereas priors on $\boldsymbol{\theta}$ typically reflect acceptable ranges of calibration parameters. Beyond these assumptions, the

choice of the prior density for $\boldsymbol{\theta}$ and \mathbf{q} must be made on a case by case basis. In Section 4, we present a particular prior distribution for $(\boldsymbol{\theta}, \mathbf{q})$ in the context of an industrial case study.

A key step in our source inversion approach is approximating the posterior probability \mathbb{P}_{post} through an approximation for the map \mathcal{A} . Let A be the GP emulator for \mathcal{A} as outlined in Section 2, then we can approximate the likelihood \mathbb{P}_{like} in (10) using

$$\widehat{\mathbb{P}}_{like}(\mathbf{w}|\boldsymbol{\theta}, \mathbf{q}) = \frac{1}{(2\pi \det \Sigma)^{\frac{1}{2}}} \exp\left(-\frac{1}{2} \left\| \Sigma^{-1/2} (A(\boldsymbol{\theta})\mathbf{q} - \mathbf{w}) \right\|^2\right). \quad (12)$$

Substituting this expression into Bayes' rule (7) yields the approximate posterior

$$\widehat{\mathbb{P}}_{post}(\boldsymbol{\theta}, \mathbf{q}|\mathbf{w}) = \frac{\widehat{\mathbb{P}}_{like}(\mathbf{w}|\boldsymbol{\theta}, \mathbf{q}) \mathbb{P}_{prior}(\boldsymbol{\theta}, \mathbf{q})}{\widehat{Z}(\mathbf{w})}, \quad (13)$$

where the constant \widehat{Z} is defined similar to (8). If the map A approximates \mathcal{A} closely, then we expect $\widehat{\mathbb{P}}_{post}$ to be close to \mathbb{P}_{post} . We refer the reader to [38] and references therein for a detailed analysis of the errors introduced through GP emulation of forward maps in Bayesian inverse problems.

With $\widehat{\mathbb{P}}_{post}$ in hand we may now compute point value estimators for the parameters $\boldsymbol{\theta}$ and source strengths \mathbf{q} . Common choices of point estimates are [16]:

$$\text{Maximum a posteriori:} \quad (\boldsymbol{\theta}, \mathbf{q})_{MAP} = \underset{\boldsymbol{\theta}, \mathbf{q}}{\operatorname{argmax}} \widehat{\mathbb{P}}_{post}(\mathbf{w}|\boldsymbol{\theta}, \mathbf{q}), \quad (14)$$

$$\text{Conditional mean:} \quad (\boldsymbol{\theta}, \mathbf{q})_{CM} = \int (\boldsymbol{\theta}, \mathbf{q}) \widehat{\mathbb{P}}_{post}(\boldsymbol{\theta}, \mathbf{q}|\mathbf{w}) d\boldsymbol{\theta} d\mathbf{q}, \quad (15)$$

$$\text{Maximum likelihood:} \quad (\boldsymbol{\theta}, \mathbf{q})_{ML} = \underset{\boldsymbol{\theta}, \mathbf{q}}{\operatorname{argmax}} \widehat{\mathbb{P}}_{post}(\boldsymbol{\theta}, \mathbf{q}|\mathbf{w}). \quad (16)$$

The uncertainty in the choice of point estimate $(\boldsymbol{\theta}^*, \mathbf{q}^*)$ can be assessed by computing the covariance matrix

$$\int ((\boldsymbol{\theta}, \mathbf{q}) - (\boldsymbol{\theta}^*, \mathbf{q}^*)) \otimes ((\boldsymbol{\theta}, \mathbf{q}) - (\boldsymbol{\theta}^*, \mathbf{q}^*)) \widehat{\mathbb{P}}_{post}(\boldsymbol{\theta}, \mathbf{q}|\mathbf{w}) d\boldsymbol{\theta} d\mathbf{q}.$$

The above estimators may of course also be computed using the true posterior \mathbb{P}_{post} , but if $\widehat{\mathbb{P}}_{post}$ is close to \mathbb{P}_{post} in an appropriate sense (such as the total variation metric) then one expects the point estimators under $\widehat{\mathbb{P}}_{post}$ to approximate their counterparts under \mathbb{P}_{post} [38].

In general, the integrals involved in computing point estimates or covariances are not analytically tractable and so it is necessary to resort to numerical methods for their estimation. Since these are high-dimensional integrals, quadrature-based approaches are unsuitable, thus it is necessary to use Markov chain Monte Carlo (MCMC) integration techniques. In this paper we use the adaptive MCMC algorithm of [9], which is outlined below in Algorithm 1. Specific values of the algorithmic parameters $(\beta_1, \gamma_1, \gamma_2, \gamma_3)$ that are tailored to our case study are provided later in Section 4.3.

4 An industrial case study

In this section we apply the Bayesian framework outlined earlier to the study of dispersion of airborne zinc particles from four point sources located within the area surrounding a lead-zinc smelter in Trail, British Columbia, Canada. Our ultimate goal is to estimate the contribution that each source makes to the total zinc released into the atmosphere by the smelting operation. We have access to monthly cumulative measurements of zinc particulate depositions at nine separate locations as well as horizontal wind field velocity data at a meteorological station located near the sources. We also have access to engineering estimates of the yearly-averaged emission rates obtained from independent engineering studies based on process control arguments. An aerial photograph of the industrial site, showing the locations of all sources and measurement devices, is provided in Figure 1. The sources are labelled q_1 to q_4 , and deposition measurements are designated R_1 to R_9 . This same emission scenario has been studied using a linear least squares approach based on a Gaussian plume approximation [22], and also by performing a finite volume discretization of the governing equation (3) [12].

We apply the atmospheric dispersion model (3), as adapted in [12] for this specific zinc smelter site with $n = 4$ sources:

$$\frac{\partial C(\mathbf{x}, t)}{\partial t} + \mathbf{L}(p, z_0, z_i, L, z_{cut})C(\mathbf{x}, t) = \sum_{j=1}^4 q_j \delta(\mathbf{x} - \mathbf{x}_j). \quad (17)$$

Algorithm 1 Adaptive Metropolis-Hastings Algorithm.

-
- 1: Choose $(\boldsymbol{\theta}_1, \mathbf{q}_1) \in \mathbb{R}^d$ in the support of $\mathbb{P}_{post}(\boldsymbol{\theta}, \mathbf{q}|\mathbf{w})$ and fixed parameters $\beta \in (0, 1)$, $\gamma_1, \gamma_2, \gamma_3 \in (0, \infty)$, $N \in \mathbb{N}$.
 - 2: **for** $j = 2 : N$ **do**
 - 3: **if** $j \leq 2d$ **then**
 - 4: Draw u from $\mathcal{N}((\boldsymbol{\theta}_j, \mathbf{q}_j), \frac{\gamma_1}{d} I_{d \times d})$.
 - 5: **else**
 - 6: Estimate the empirical covariance matrix Σ_j using $\{(\boldsymbol{\theta}_k, \mathbf{q}_k)\}_{k=1}^j$.
 - 7: Draw u from $(1 - \beta)\mathcal{N}((\boldsymbol{\theta}_j, \mathbf{q}_j), \frac{\gamma_2}{d} \Sigma_j) + \beta\mathcal{N}((\boldsymbol{\theta}_j, \mathbf{q}_j), \frac{\gamma_3}{d} I_{d \times d})$.
 - 8: **end if**
 - 9: Propose $(\tilde{\boldsymbol{\theta}}_j, \tilde{\mathbf{q}}_j) = (\boldsymbol{\theta}_{j-1}, \mathbf{q}_{j-1}) + u$.
 - 10: Compute $\delta = \min\left(1, \frac{\mathbb{P}_{post}(\tilde{\boldsymbol{\theta}}_j, \tilde{\mathbf{q}}_j|\mathbf{w})}{\mathbb{P}_{post}(\boldsymbol{\theta}_{j-1}, \mathbf{q}_{j-1}|\mathbf{w})}\right)$.
 - 11: Draw w from $U([0, 1])$.
 - 12: **if** $w < \delta$ **then**
 - 13: $(\boldsymbol{\theta}_j, \mathbf{q}_j) = (\tilde{\boldsymbol{\theta}}_j, \tilde{\mathbf{q}}_j)$ (Accept the move)
 - 14: **else**
 - 15: $(\boldsymbol{\theta}_j, \mathbf{q}_j) = (\boldsymbol{\theta}_{j-1}, \mathbf{q}_{j-1})$ (Reject the move)
 - 16: **end if**
 - 17: **end for**
-

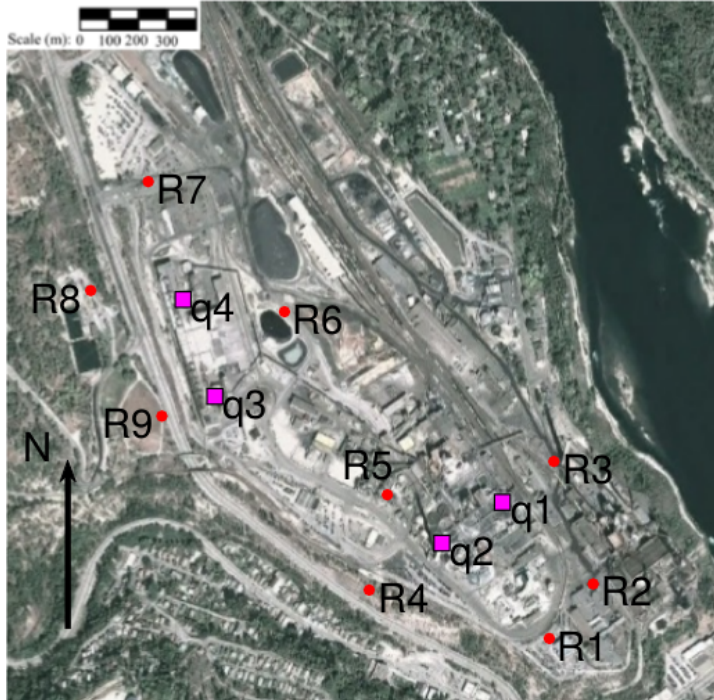


Fig. 1: Aerial photograph of the lead-zinc smelter in Trail, BC, Canada. The square points labelled 'qn' represent the locations of the four zinc sources, while the circular points 'Ri' denote the nine measurement devices.

The differential operator \mathbf{L} depends non-linearly on five parameters as we shall explain next. The parameter p comes from the assumption of a power-law distribution in the vertical component of the wind profile \mathbf{v} ; that is, if z denotes the height above ground level $(x, y, 0)$, then the two horizontal velocity components v_x and v_y are assumed to be function of z and t only and satisfy

$$\|(v_x(z, t), v_y(z, t))\|_2 = v_r(t) \left(\frac{z}{z_r}\right)^p,$$

where $v_r(t)$ is the wind speed at some reference height z_r . The exponent p depends on factors such as surface roughness and atmospheric stability class. For more details about the power law model for the wind velocity the reader is referred to [34]. The other four parameters are related to the entries in the diagonal diffusivity matrix $\mathbf{D} = \text{diag}(D_{11}, D_{22}, D_{33})$ in equation (3). Following [34], the vertical diffusion

coefficient satisfies to a good approximation

$$D_{33} = \frac{\kappa v_* z}{\phi(z/L)}, \quad (18)$$

where κ is the *von Karman* constant whose value can be set to 0.4 in practical scenarios. The function ϕ in the denominator is taken to be a piecewise continuous function

$$\phi\left(\frac{z}{L}\right) = \begin{cases} 1 + 4.7\frac{z}{L}, & \text{if } \frac{z}{L} \geq 0, \\ (1 - 15\frac{z}{L})^{-\frac{1}{2}}, & \text{if } \frac{z}{L} < 0, \end{cases}$$

where L is called the *Monin-Obukhov length*. The parameter v_* is known as the *friction velocity* and is represented by

$$v_*(t) = \frac{\kappa v_r(t)}{\ln\left(\frac{z_r}{z_0}\right)},$$

where $v_r(t)$ is the wind velocity at some reference height. The variable z_0 is called the *roughness length* and depends on both terrain and surface type. The remaining horizontal diffusion coefficients satisfy $D_{11} = D_{22}$ and are assumed to be independent of height z . A correlation that is commonly employed for these parameters is [34]

$$D_{11} = D_{22} \approx \frac{v_* z_i^{\frac{3}{4}} (-\kappa L)^{-\frac{1}{3}}}{10},$$

where the parameter z_i is called the *mixing layer height*.

The governing PDE (17) is complemented with boundary conditions defined over the space-time domain $\mathbb{R}^2 \times [0, \infty) \times (0, T)$ for some fixed $T \in \mathbb{R}$. For this purpose, we impose the following conditions at infinity

$$c(\mathbf{x}, t) \rightarrow 0 \quad \text{as } \|\mathbf{x}\| \rightarrow \infty,$$

along with a flux (or Robin) boundary condition at the ground surface

$$\left(v_{set} c + D_{33} \frac{\partial c}{\partial z} \right) \Big|_{z=0} = v_{dep} c \Big|_{z=0}, \quad (19)$$

where $v_{set} = 0.0027 \text{ m/s}$ is the settling velocity of zinc particles and $v_{dep} = 0.005 \text{ m/s}$ is the corresponding deposition velocity. Observe that at the bottom boundary $z = 0$, the vertical diffusivity satisfies $D_{33}(0) = 0$ from (18) which leads to an inconsistency in the Robin boundary condition; therefore, we define a cut-off length $z_{cut} \gtrsim 0$ below which D_{33} is set to the non-zero constant value $D_{33}(z_{cut})$.

Equation (17) is solved numerically using the finite volume algorithm described [12]. The code takes as input values of the wind speed $v_r(t)$, parameters p, z_0, z_i, L, z_{cut} , and source strengths q_j , and returns as output the zinc concentration at any point (x, y, z) and time $0 < t < T$. However, as mentioned earlier, we do not have experimental measurements of concentration but rather total depositions at the nine sites R_1, \dots, R_9 (see Figure 1). We model the depositions using the integral

$$w(x, y, T) = \int_0^T c(x, y, 0, t) v_{set} dt, \quad (20)$$

where v_{set} is the settling velocity of zinc particles given by Stokes' law [34]. Since the nine deposition measurements R_i were obtained from roughly cylindrical dustfall jar collectors, we can readily approximate the total deposition over the time interval $(0, T]$ at site $i = 1, \dots, 9$ using

$$w_i = \int_{jar} w(x, y, T) dx dy \approx w(x_i, y_i, T) \Delta A, \quad (21)$$

where $\Delta A \approx 0.0206 \text{ m}^2$ is the cross-sectional area of each dustfall jar. We then collect the zinc deposition measurements into a vector $\mathbf{w} = (w_i)^T$ and from now on, we assume that the time interval T over which particulates are allowed to accumulate in the dustfall jars is one month.

With the industrial case fully specified our source inversion framework can be implemented. We make the following observations:

- The vector \mathbf{w} is composed of $d = 9$ measurements, one from each jar, so that $\mathbf{w} \in \mathbb{R}^9$.
- The number of model parameters is $m = 5$, which are represented by the vector $\boldsymbol{\theta} = (p, z_0, z_i, L, z_{cut})^T$.
- There are $n = 4$ sources stored in the vector $\mathbf{q} = (q_1, q_2, q_3, q_4)^T$.

Each measurement in the vector \mathbf{w} may then be viewed as a map

$$w_i : \mathbb{R}^5 \times \mathbb{R}^4 \rightarrow [0, \infty).$$

In other words, the number $w_i(\boldsymbol{\theta}^*, \mathbf{q}^*)$ is the deposition at site R_i predicted by the finite volume solver described in [12], given a particular choice of parameters $\boldsymbol{\theta}^*$ and source strengths \mathbf{q}^* . In the framework of Section 2 (namely, equations (4) and (5)) we write

$$w_i(\boldsymbol{\theta}^*, \mathbf{q}^*) = \sum_{j=1}^4 \mathbf{a}_{ij}(\boldsymbol{\theta}^*) q_j^*, \quad (22)$$

where each of the nonlinear functions $\mathbf{a}_{ij}(\boldsymbol{\theta}^*)$ can be approximated using a GP emulator $a_{ij}(\boldsymbol{\theta}^*)$.

4.1 Sensitivity analysis

Before solving the source inversion problem, we first perform a sensitivity analysis in order to check whether the dimensionality of the parameter space can be reduced. The goal of a sensitivity analysis is to assess the relative importance of each of the variables in a model [32]. In this paper, the sensitivity measure we employ is known as the Sobol total index [32, 36, 39] which captures the variance of the model outputs due to changes in each of its inputs and is a measure of the relative importance of each input. More specifically, the Sobol total index ranges from a lower limit of 0, denoting a variable that has no effect on the output, to an upper limit of 1, indicating that all variability in the model is explained by the variable in hand.

We perform our sensitivity analysis using the R package `sensitivity` [28], which uses Gaussian processes as surrogates for the function of interest. In order to account for the variability introduced by using different isotropic kernels (see Definition 1), we calculate Sobol indices using the following choices of the kernel κ :

- Exponential: $\kappa(|\boldsymbol{\theta} - \boldsymbol{\theta}'|) = r_1 \exp\left(-\frac{|\boldsymbol{\theta}' - \boldsymbol{\theta}|}{r_2}\right)$,
- Squared exponential: $\kappa(|\boldsymbol{\theta} - \boldsymbol{\theta}'|) = r_1 \exp\left(-\frac{|\boldsymbol{\theta} - \boldsymbol{\theta}'|^2}{2r_2}\right)$,
- Matérn $\frac{3}{2}$: $\kappa(|\boldsymbol{\theta} - \boldsymbol{\theta}'|) = r_1 \left(1 + \frac{\sqrt{3}|\boldsymbol{\theta} - \boldsymbol{\theta}'|}{r_2}\right) \exp\left(-\frac{\sqrt{3}|\boldsymbol{\theta} - \boldsymbol{\theta}'|}{r_2}\right)$,
- Matérn $\frac{5}{2}$: $\kappa(|\boldsymbol{\theta} - \boldsymbol{\theta}'|) = r_1 \left(1 + \frac{\sqrt{5}|\boldsymbol{\theta} - \boldsymbol{\theta}'|}{r_2} + \frac{5}{3} \left(\frac{|\boldsymbol{\theta} - \boldsymbol{\theta}'|}{r_2}\right)^2\right) \exp\left(-\frac{\sqrt{5}|\boldsymbol{\theta} - \boldsymbol{\theta}'|}{r_2}\right)$.

In the above, $r_1, r_2 \in \mathbb{R}$ are tuning parameters. These kernels are commonly used in Gaussian process regression and are readily implemented in the R packages `sensitivity` and `DiceKriging`. For other possible choices of kernels see [31, Sec. 4]

The sensitivity indices computed using the four different kernels are summarized in the boxplots shown in Figure 2. The size of each box represents the interquartile range (IQR). The lower bar attached to each box is at the value

$$\min\left\{\max(S), E_3 + \frac{3}{2}IQR\right\},$$

and the upper bar is located at

$$\max\left\{\min(S), E_1 - \frac{3}{2}IQR\right\},$$

where S is the set of the four indices (corresponding to the four kernels) calculated for each w_i for $i \in \{1, \dots, 9\}$, and E_1 and E_3 are the first and third quartiles of S respectively [23].

From these results it is clear that the variability in output of each map w_i is captured in large part by the two parameters p and z_0 . Although the sensitivity index for L is relatively small, we still retain this parameter because it is known to be closely related to z_0 [34, Chap. 19]. The remaining “unimportant” parameters, z_i and z_{cut} , may then be fixed at suitable constant values (we used $z_i = 100$ and $z_{cut} = 2$, following [12]). This reduces the dimensionality of the parameter space from 5 to 3 and allows us to redefine our parameter vector as

$$\boldsymbol{\theta} := (p, z_0, L).$$

We next identify the support of the prior distribution, which consists simply of the interval of allowable values for each unknown parameter. Every emission rate must be a non-negative number, so that

$$q_j \in [0, \infty), \quad \text{for } j = 1, 2, 3, 4.$$

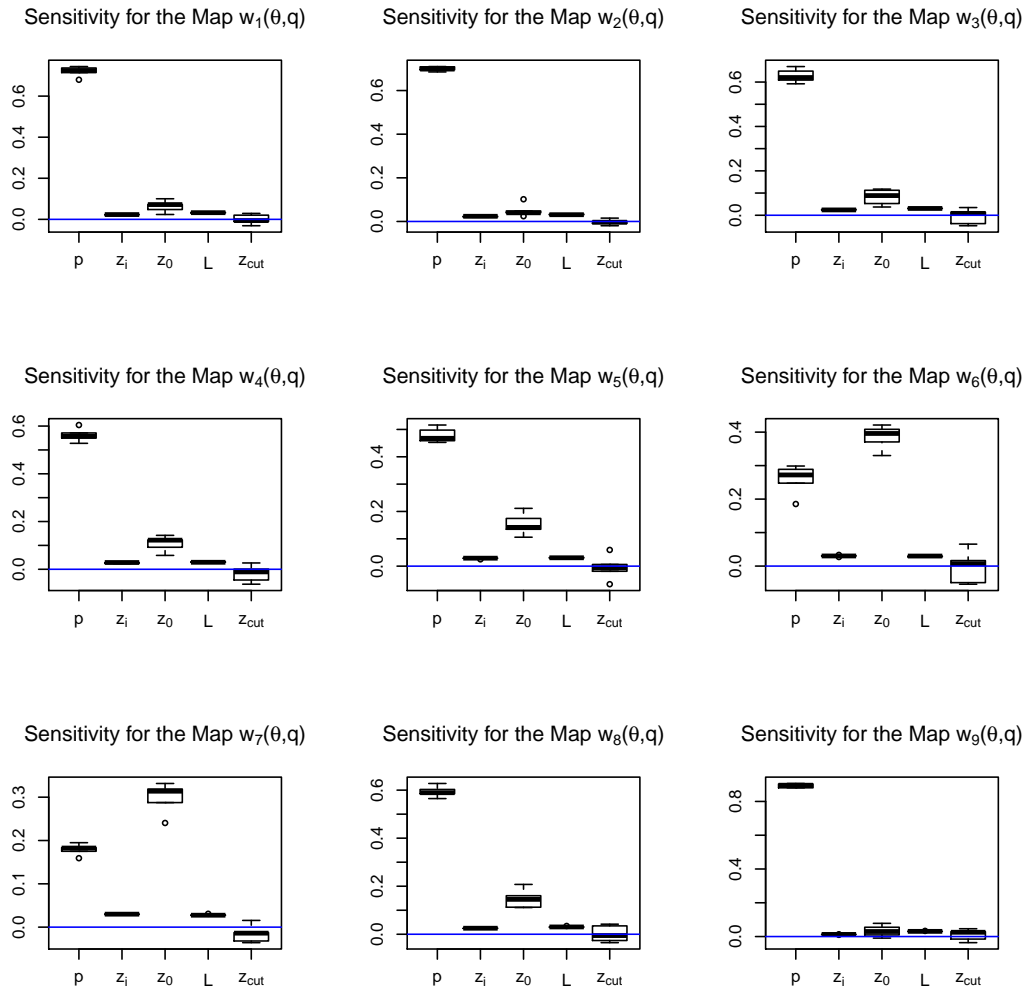


Fig. 2: The Sobol total indices of the model parameters θ for the maps w_i defined in (22). Errors bars indicate variation between estimated value of Sobol total indices using different kernels.

For the remaining parameters p, z_0, L , we use the physically-motivated ranges listed in Table 1, which are slightly larger than the suggested ranges typically cited in the literature (see e.g., [34, Ch. 19]) to allow for more flexibility. Consequently, each map $w_i(\theta, \mathbf{q})$ has a domain

$$\text{dom}(w_i) = [0, 0.6] \times [0, 3] \times [-600, 0] \times [0, \infty)^4.$$

With this information in hand, we are now in a position to create a space-filling design needed to construct the GP emulator as explained in Section 2. Due to computational budget constraints, we choose 64 points to construct our emulators. We display 2D projections of the corresponding maximin space-filling design for the three relevant parameters in Figure 3.

Table 1: Physically reasonable ranges for the parameters p, z_0, L .

Parameter (units)	Symbol	Range
Velocity exponent	p	$[0, 0.6]$
Roughness length (m)	z_0	$[0, 3]$
Monin-Obukhov length (m)	L	$[-600, 0]$

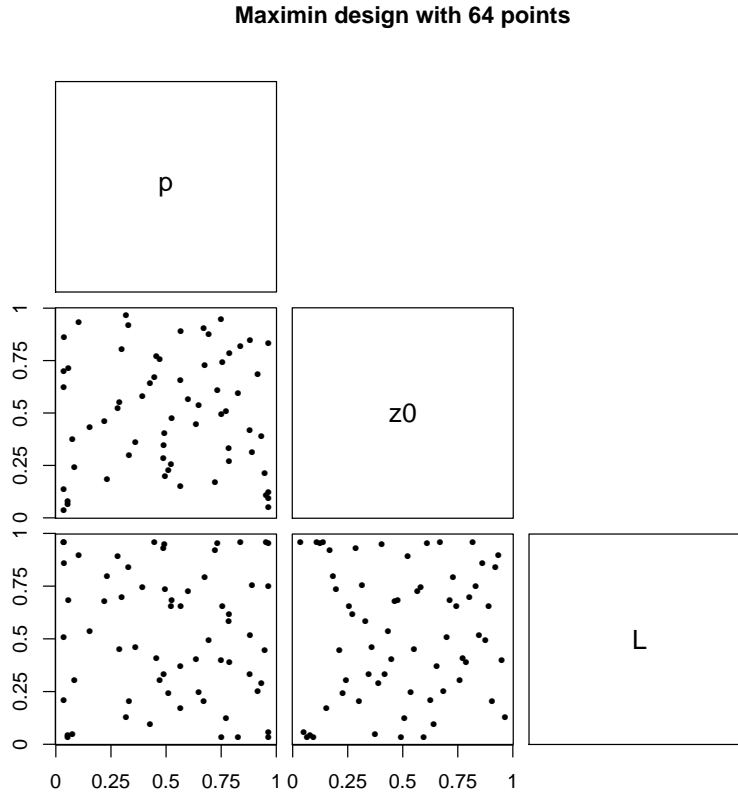


Fig. 3: Pairwise two-dimensional projections of the space-filling design with 64 points for emulating \mathcal{A} . Results were obtained by initializing the particle swarm algorithm using a LHD and iterating for 10,000 steps. The range of each parameter is normalized between 0 and 1 for easy visualization. The scatter plot in the second row depicts z_0 vs. p . The plots on the bottom row show L vs. p (left) and L vs. z_0 (middle).

4.2 Construction and validation of the emulator

We now proceed as outlined in Section 2 to construct the emulator for maps \mathbf{a}_{ij} using Gaussian processes (GP). The 64 different combinations of parameters p, z_0, L obtained from the space-filling design in the previous section are used to train the GP. To create the GPs for each of the 36 maps $\mathbf{a}_{ij}(p, z_0, L)$, we used the R package `DiceKriging` [5]. Unlike the sensitivity analysis in the previous sections where four different kernels were computed, the GPs here are constructed using only the isotropic squared exponential kernel.

The `DiceKriging` package chooses a maximum likelihood estimate for the tuning parameters r_1, r_2 for each map \mathbf{a}_{ij} . To assess the quality of each emulator we perform a “leave one out” cross validation (LOOCV) [25, Sec 1.4] on each of the nine sites R_1, R_2, \dots, R_9 for the 64 design points shown in Figure 3; that is, we run the finite volume solver 64 times and save the predicted deposition value for each sites, yielding a total of 64×9 points or predictions from the finite volume solver. For each point “left out” in the LOOCV, we plot in Figure 4 the deposition obtained from the finite volume solver versus the predicted value from the corresponding trained GP. This figure shows that the output of the GP is closely matched to the output of the finite volume solver at most design points, with the exception of a small number of outliers.

4.3 Source inversion

The GP emulator may now be employed to estimate the emission rates q_1, \dots, q_4 , for the actual physical dataset which consists of dustfall jar measurements at the nine locations marked R_1, \dots, R_9 in Figure 1. Our first task is to specify the prior distribution for $\boldsymbol{\theta}$ and \mathbf{q} . Following (11), we take the prior for p, z_0, L to be independent of the prior on \mathbf{q} . Since we do not have any prior information about the values of the parameters p, z_0, L (other than an acceptable range), we choose a uniform distribution for each parameter

$$\mathbb{P}_{\text{prior}}(p) \propto \mathbf{1}_{[0,0.6]}(p), \quad \mathbb{P}_{\text{prior}}(z_0) \propto \mathbf{1}_{[0,3]}(z_0), \quad \mathbb{P}_{\text{prior}}(L) \propto \mathbf{1}_{[-600,0]}(L),$$

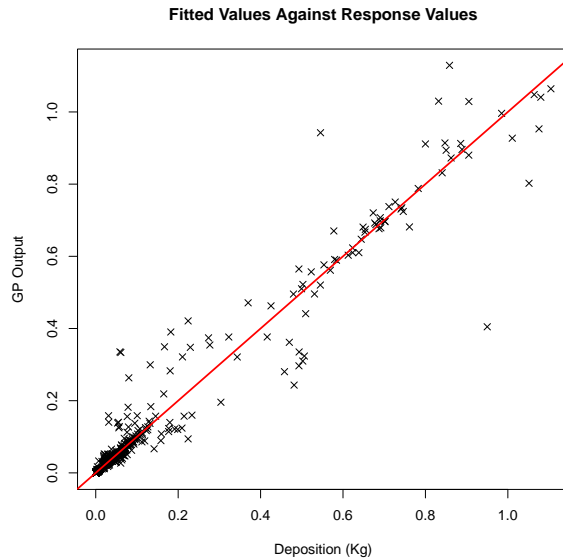


Fig. 4: Comparison between deposition values predicted by the finite volume solver and the GP emulator using LOOCV for each of the 64×9 data points obtained from the finite volume solver.

where $\mathbf{1}_E$ is the indicator function for the interval set E . For the source strengths \mathbf{q} , we have the engineering estimates from previous independent studies shown in Table 2 which can be applied along with a non-negativity constraint to construct a suitable prior. In particular, we propose the following prior for source strengths based on a gamma distribution:

$$q_j \propto q_j^{\alpha_j - 1} \exp(-\beta_j q_j).$$

The quantities α_j are called shape parameters, while the β_j are rate parameters for the gamma distribution. Values for α_j and β_j are derived from the engineering estimates $\mathbf{q} = (q_{eng,j})$ as the solution to the following system of equations:

$$\frac{\alpha_j - 1}{\beta_j} = q_{eng,j},$$

$$qgamma(0.99, \alpha_j, \beta_j) = 3q_{eng,j}.$$

Here, $qgamma$ is the *quantile function*, which is just the inverse of the cumulative distribution function of the gamma density. Choosing α_j and β_j in this way guarantees that the mode of the prior distribution for q_j is located at the corresponding engineering estimate. Furthermore, 99% of the prior mass is concentrated in the interval $(0, 3q_{eng,j})$ for each q_j [8, Sec. 2.2].

Table 2: Engineering estimates of the source strengths from previous independent studies.

Source	Estimated emission rate [ton/yr]
$q_{eng,1}$	35
$q_{eng,2}$	80
$q_{eng,3}$	5
$q_{eng,4}$	5

Our next task is to specify the covariance matrix Σ for the measurement noise in (9). Because the dustfall jars are well-separated, we assume that the measurement noise for each jar is independent from the others and hence take

$$\Sigma = \lambda I_{9 \times 9}, \quad (23)$$

where $\lambda > 0$ is the variance of the measurement noise and $I_{9 \times 9}$ is the 9×9 identity matrix. The variance parameter λ is related to the signal-to-noise ratio (SNR) of the measurements and can be estimated by

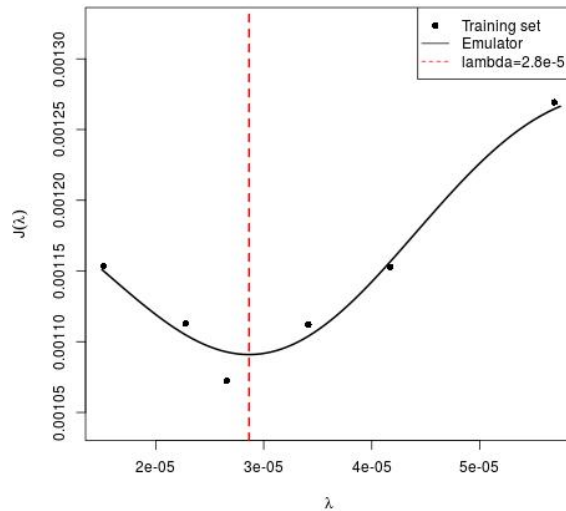


Fig. 5: Emulator for $J(\lambda)$ in equation (24).

minimizing the functional

$$J(\lambda) = \frac{1}{2} \int \left(\left\| \hat{A}(p, L, z_0) \mathbf{q} - \mathbf{w} \right\|_2 + \left\| \mathbf{q} - \mathbf{q}_{eng} \right\|_2 \right) d\mathbb{P}_{post}^\lambda, \quad (24)$$

where $\mathbf{q}_{eng} = [35, 80, 5, 5]^T \text{ ton/yr}$, and the notation $\mathbb{P}_{post}^\lambda$ is used to explicitly show the dependence of the posterior measure \mathbb{P}_{post} on the parameter λ . The motivation for defining $J(\lambda)$ in this way comes from the problem of choosing λ to minimize the average value of the expression

$$(1 - \delta) \|A(p, L, z_0) \mathbf{q} - \mathbf{w}\|_2 + \delta \|\mathbf{q} - \mathbf{q}_{eng}\|_2 \quad \text{for } \delta \in [0, 1]. \quad (25)$$

Depending on the value of δ , different weight can be given to the credibility of the atmospheric dispersion model compared to that of prior information about \mathbf{q} . In particular, if $\delta = 1$ then (25) reduces to $\|\mathbf{q} - \mathbf{q}_{eng}\|_2$ and λ is chosen to solely match the engineering estimates; on the other hand, taking $\delta = 0$ leaves only the term $\|A(p, L, z_0) \mathbf{q} - \mathbf{w}\|_2$ which chooses λ solely based on minimizing the data misfit. We choose $\delta = \frac{1}{2}$, which is a compromise between the two extremes. After substituting into (25) and minimizing the expected value with respect to the posterior measure, the expression $J(\lambda)$ from (24) is easily obtained.

Computing $J(\lambda)$ analytically is not practically feasible and so we approximate it using a GP emulator, minimizing the emulator instead. To obtain design points for the GP, we evaluate J for six different values of λ using Markov Chain Monte Carlo integration, choosing only six points to keep the computational cost within reasonable limits. The design points were chosen empirically by identifying locations where the function J changes the most, and the resulting simulation is shown in Figure 5. The minimum value of $J(\lambda)$ is attained at $\lambda \approx 2.8 \times 10^{-5}$, which is the value of λ we use in our source inversion framework for (23). Note that optimization with emulators is an emerging area of research for which a detailed background can be found in the paper [27] and references therein.

We now apply Algorithm 1 to generate samples from the posterior distribution, using the GP emulator $A(p, z_0, L)$ to compute the likelihood ratio in Step 10 and parameter values listed in Table 3. When estimating integrals, we discard the first half of the samples (up to 500,000) and retain only every tenth iterate from there onwards to reduce correlation between consecutive samples. Therefore, a total of 50,000 samples is used to approximate each integral with respect to the posterior.

Marginal posterior histograms are displayed in Figure 6, from which we observe that marginals for the emission rates q_j are skewed and have well-defined modes. This is in line with our choice of a gamma prior, and the mode for these marginals is a reasonable point estimate for each q_j . To approximate the marginal mode we fit a gamma distribution to the histograms and take the point estimate to be the mode of the fitted distribution.

The situation is slightly different for the parameters p, z_0, L . The histogram for p has no distinctive maximum point indicating that the data is not informative for this parameter. For z_0 and L , some values are more distinctive than others but the distinction is still not sharp. To choose a point estimate for these

Table 3: Value of parameters used in Algorithm 1 for generating samples from the posterior distribution.

N	β	γ_1	γ_2	γ_3
10^6	0.05	0.01	$(2.38)^2$	$(0.1)^2$

cases, we first fit a density over the histograms and then take the resulting mode as the point estimate, which is indicated in each case by a red dashed line in Figure 6.

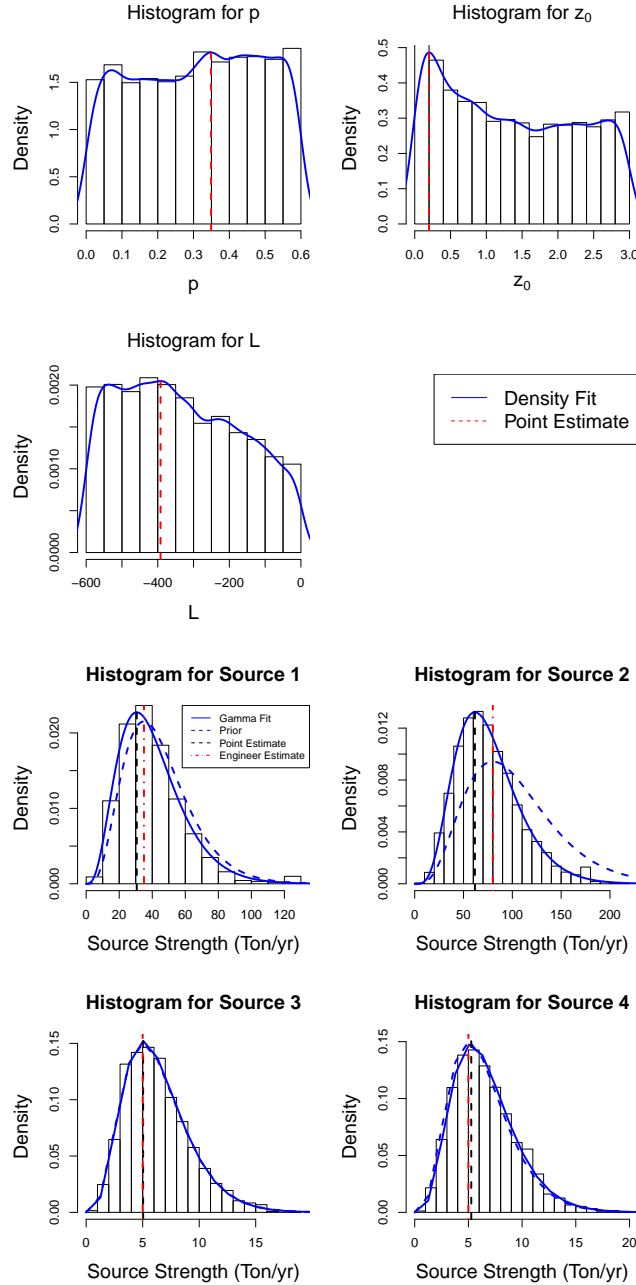


Fig. 6: Marginal posterior distribution for the three parameters p , z_0 , L (top) and the four emission rates q_1 , q_2 , q_3 , q_4 (bottom).

To quantify the uncertainty in each parameter, we employ a 68% Bayesian confidence interval which is defined as follows: given a point estimate x^* of a random variable X distributed with probability density ρ , a 68% Bayesian confidence interval is the radius of the ball centered at x^* containing 68% of the probability mass for ρ . The corresponding confidence interval is listed for each parameter in Table 4. The estimates

of the q_j are clearly more informative, which is due to the linearity of the forward map and stronger prior knowledge. The marginal modes and associated uncertainties for the q_j are depicted graphically in Figure 7. These results suggest that the engineering estimates for sources q_1 and q_2 are most likely overestimates, whereas sources q_3 and q_4 are slightly underestimated. Furthermore, our estimates for sources q_2 , q_3 and q_4 qualitatively agree with the results of previous studies in [22] (L&S) and [12] (H&S).

The results obtained in (L&S) were obtained using a Gaussian plume model and a least-squares based inversion method. Therefore, discrepancies between our estimates and those of (L&S) are expected given that we used a very different approach. On the other hand, (H&S) used the same PDE based forward map as this study (although within a different Bayesian formulation) and so it is more informative to compare our results to those of (H&S). We note two key differences between our results. First of all, there is a notable difference between the estimates for q_1 . In [12], the parameters p, z_0, L are fixed at 0.2, 0.05 and -10 , respectively. But these values depend strongly on environmental conditions, so a mis-assessment in the environmental conditions might lead to sub-optimal predictions. It was conjectured in [12] that this may be the underlying reason for the apparent overestimation of q_1 in comparison to previous studies. Furthermore, (H&S) obtained their estimates by marginalizing the posterior measure on q_j only, whereas we marginalize the posterior measure on the q_j and the model parameters θ .

The second major difference is that our estimates have larger error bars in comparison to (H&S), although we note that our error bars are large enough to cover most previous estimates. This is largely due to differences in the parameter λ which have a direct effect on posterior variance. We chose λ by minimizing $J(\lambda)$ in (24) and achieved an SNR of 0.37. This means that the strength of the signal is quite low compared to the noise of the measurements. In [12], λ is chosen to achieve an SNR of 10 which is why the error bars for those results are much smaller. Furthermore, our method accounts for uncertainties in the value of the model parameters θ while these are fixed in the approach of (H&S).

We further observe that our estimate of the total emission is consistent with previous studies, with the exception of (H&S) which appears to have a higher estimate. This is mostly due to the over-estimation of q_1 which, as was mentioned before, is likely due to a lack of calibration in [12]. By allowing the model parameters to be calibrated automatically we have obtained estimates that are consistent with two previous studies that used very different methodologies, namely the engineering estimates and (L&S). This highlights the importance of model calibration in atmospheric source inversion studies. Furthermore, our uncertainty estimates are more realistic since we account for model parameter uncertainties and infer a better value of the signal-to-noise ratio. Although our uncertainty estimate for the total emission is relatively large in comparison to the estimate itself, this is due to the fact atmospheric source inversion problems are severely ill-posed and sensitive to model parameters.

Table 4: Parameters and their estimates and uncertainties.

Parameter	Point Estimate	68% Confidence Interval
p	0.3478	[0.1498, 0.5458]
z_0	0.0811	[0, 1.5781]
L	-379.45	[-195.86, -563.04]

Finally, we note that Gaussian process emulation is a crucial aspect of our inversion method that renders it cost-effective. The main computational bottleneck in our framework is the evaluations of the finite volume forward solver at the 64 design points used to construct the emulator. Given that the design points are independent not only of each other but also the measurements, these computations can be performed offline and in parallel. Every run of the finite volume code requires roughly two hours on a personal computer, while the emulator can be evaluated in a fraction of a second. Therefore, the cost of evaluating the emulator is negligible in comparison to the finite volume solver which enables us to perform Monte Carlo integration in a reasonable time. Clearly, such a calculation would be prohibitive had we employed the finite volume solver directly in the MCMC step.

In comparison to the previous approaches of [12] and [22] our total computational cost (including the training of the emulator and design of experiment) is higher. For example, [22] uses a Gaussian plume model which is much cheaper than the finite volume solver but comes at the cost of major simplifications to the physics. On the other hand, [12] uses a finite volume forward solver of the original PDEs, which is only evaluated four times. However, neither of these two approaches provides an automated calibration procedure, and so we claim that the extra computations in our methodology are a small price to pay for a richer solution structure with more realistic uncertainty estimates and rigorous model calibration.

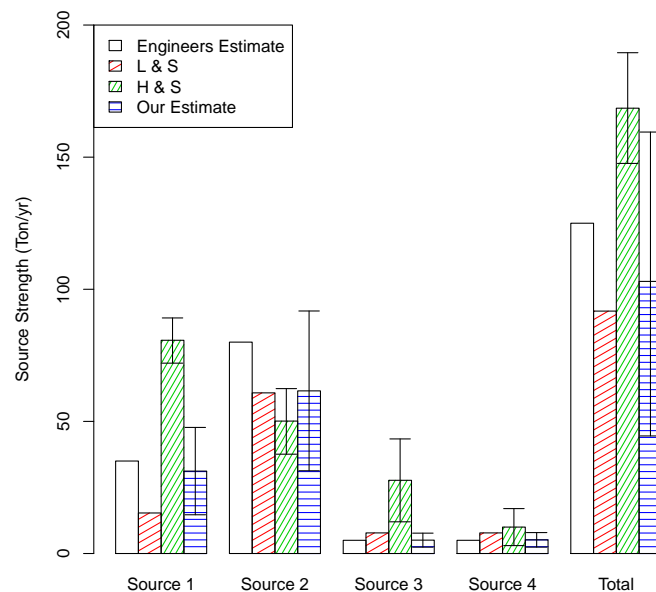


Fig. 7: Comparison between engineering estimates of the emission rates and our point estimates, along with the associated uncertainty. The “Total” column indicates the sum of emission rates from all four sources. Previous results from Lushi and Stockie [22] (L&S) and Hosseini and Stockie [12] (H&S) are included for comparison purposes.

5 Conclusions

We have developed a new method for simultaneously calibrating atmospheric dispersion models and solving source inversion problems. Gaussian process emulators are used to speed up the evaluation of the atmospheric dispersion model, which are then employed within a Bayesian framework to calibrate model parameters and estimate source strengths at the same time. This allows for automatic tuning of unknown model parameters rather than more heuristic approaches that are typically used in practice.

We demonstrated the effectiveness of our proposed method in an industrial case study concerning emissions of zinc from a lead-zinc smelter in Trail, BC, Canada. Our results agree with those of previous studies in the literature that are based on very different approaches. Hence, we conclude that the emulation process is an accurate and efficient one that has the added advantage of greatly speeding up the calculations by exploiting emulation.

Our proposed solution methodology is not restricted to atmospheric dispersion models governed by partial differential equations. Indeed, our method can be adapted to any application where one is dealing with a forward map that is linear in the primarily unknowns (such as the source strengths) but may depend nonlinearly on the model parameters.

Acknowledgements This work was partially supported by the Natural Sciences and Engineering Research Council of Canada through a Postdoctoral Fellowship (BH) and a Discovery Grant (JMS). We are grateful to the Environmental Management Group at Teck Resources Ltd. (Trail, BC) for providing data and for many useful discussions.

References

1. R. K. Arora. *Optimization: Algorithms and Applications*. CRC Press, Boca Raton, FL, 2015.
2. P. C. Chatwin. The use of statistics in describing and predicting the effects of dispersing gas clouds. *Journal of Hazardous Materials*, 6(1-2):213–230, 1982.
3. S. Conti, J. P. Gosling, J. E. Oakley, and A. O’Hagan. Gaussian process emulation of dynamic computer codes. *Biometrika*, 96(3):663–676, 2009.
4. Cooper Environmental, Beaverton, OR. *Ambient Monitoring: Xact 625 (product description)*, 2015. <http://cooperenvironmental.com/ambient-monitoring>.
5. D. Dupuy, C. Helbert, J. Franco, et al. DiceDesign and DiceEval: two R packages for design and analysis of computer experiments. *Journal of Statistical Software*, 65(11):1–38, 2015.

6. I. G. Enting and G. N. Newsam. Inverse problems in atmospheric constituent studies: II. Sources in the free atmosphere. *Inverse Problems*, 6(3):349–362, 1990.
7. K.-T. Fang, D. K. J. Lin, P. Winker, and Y. Zhang. Uniform design: Theory and application. *Technometrics*, 42(3):237–248, 2000.
8. W. Gilchrist. *Statistical Modelling with Quantile Functions*. CRC Press, Boca Raton, FL, 2000.
9. H. Haario, E. Saksman, and J. Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242, 2001.
10. S. E. Haupt and G. S. Young. Paradigms for source characterization. In *Proceedings of the 15th Joint Conference on the Applications of Air Pollution Meteorology with A&WMA*, page J6.1, New Orleans, LA, Jan. 20–24, 2008. American Meteorological Society.
11. B. Hosseini and J. M. Stockie. Bayesian estimation of airborne fugitive emissions using a Gaussian plume model. *Atmospheric Environment*, 141:122–138, 2016.
12. B. Hosseini and J. M. Stockie. Estimating airborne particulate emissions using a finite-volume forward solver coupled with a Bayesian inversion approach. *Computers & Fluids*, 154:27–43, 2017.
13. V. Isakov. *Inverse Source Problems*, volume 34 of *Mathematical Surveys and Monographs*. American Mathematical Society, Providence, RI, 1990.
14. M. E. Johnson, L. M. Moore, and D. Ylvisaker. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, 26(2):131–148, 1990.
15. B. Jones and R. T. Johnson. Design and analysis for the Gaussian process model. *Quality and Reliability Engineering International*, 25(5):515–524, 2009.
16. J. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*, volume 160 of *Applied Mathematical Sciences*. Springer Science & Business Media, New York, NY, 2005.
17. A. Keats, E. Yee, and F.-S. Lien. Bayesian inference for source determination with applications to a complex urban environment. *Atmospheric Environment*, 41(3):465–479, 2007.
18. M. C. Kennedy and A. O’Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society, Series B – Statistical Methodology*, 63(3):425–464, 2001.
19. Á. Leelőssy, F. Molnár, F. Izsák, Á. Havasi, I. Lagzi, and R. Mészáros. Dispersion modeling of air pollutants in the atmosphere: A review. *Open Geosciences*, 6(3):257–278, 2014.
20. W. S. Lewellen and R. I. Sykes. Meteorological data needs for modeling air quality uncertainties. *Journal of Atmospheric and Oceanic Technology*, 6(5):759–768, 1989.
21. C.-H. Lin and L.-F. W. Chang. Relative source contribution analysis using an air trajectory statistical approach. *Journal of Geophysical Research: Atmospheres*, 107(D21):ACH 6–1–ACH 6–10, 2002.
22. E. Lushi and J. M. Stockie. An inverse Gaussian plume approach for estimating atmospheric pollutant emissions from multiple point sources. *Atmospheric Environment*, 44(8):1097–1107, 2010.
23. R. McGill, J. W. Tukey, and W. A. Larsen. Variations of box plots. *The American Statistician*, 32(1):12–16, 1978.
24. M. D. McKay, R. J. Beckman, and W. J. Conover. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.
25. K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, Cambridge, MA, 2012.
26. A. O’Hagan. Bayesian analysis of computer code outputs: A tutorial. *Reliability Engineering & System Safety*, 91(10):1290–1300, 2006.
27. M. A. Osborne, R. Garnett, and S. J. Roberts. Gaussian processes for global optimization. In *Proceedings of the 3rd International Conference on Learning and Intelligent Optimization (LION3)*, pages 1–15, 2009.
28. G. Pujol, B. Iooss, A. Janon, et al. *Sensitivity: Global Sensitivity Analysis of Model Outputs*, 2016. R package version 1.12.1.
29. K. S. Rao. Uncertainty analysis in atmospheric dispersion modeling. *Pure and Applied Geophysics*, 162(10):1893–1917, 2005.
30. K. S. Rao. Source estimation methods for atmospheric dispersion. *Atmospheric Environment*, 41:6964–6973, 2007.
31. C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.
32. A. Saltelli, K. Chan, and E. M. Scott. *Sensitivity Analysis*, volume 134 of *Wiley Series in Probability and Statistics*. Wiley, New York, NY, 2000.
33. P. Sebastiani and H. P. Wynn. Maximum entropy sampling and optimal Bayesian experimental design. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(1):145–157, 2002.
34. J. H. Seinfeld and S. N. Pandis. *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change*. John Wiley & Sons, New York, NY, 1997.
35. Y. N. Skiba. On a method of detecting the industrial plants which violate prescribed emission rates. *Ecological Modelling*, 159(2):125–132, 2003.
36. I. M. Sobol. Sensitivity estimates for non-linear mathematical models. *Mathematical Modelling and Computational Experiments*, 1(4):407–414, 1993.
37. M. D. Sohn, P. Reynolds, N. Singh, and A. J. Gadgil. Rapidly locating and characterizing pollutant releases in buildings. *Journal of the Air & Waste Management Association*, 52(12):1422–1432, 2002.
38. A. M. Stuart and A. L. Teckentrup. Posterior consistency for Gaussian process approximations of Bayesian posterior distributions. *Mathematics of Computation*, 87(310):721–753, 2018.
39. T. J. Sullivan. *Introduction to Uncertainty Quantification*. Springer, New York, NY, 2015.
40. Thermo Scientific, Waltham, MA. *High-Volume Air Samplers (product description)*, 2015. <http://www.thermoscientific.com/en/product/high-volume-air-samplers.html>.
41. D. B. Turner. *Workbook of Atmospheric Dispersion Estimates: An Introduction to Dispersion Modeling*. CRC Press, Boca Raton, FL, 1994.