Comparison of the Gaussian plume and puff atmospheric dispersion models for methane modeling on oil and gas sites

Meng Jia,* William S. Daniels, and Dorit M. Hammerling

Department of Applied Mathematics and Statistics, Colorado School of Mines, Golden, Colorado, United States

E-mail: mjia@mines.edu

Abstract

Characterizing methane emissions on oil and gas sites often relies on a forward model to describe the atmospheric transport of methane. Here we compare two forward models: the Gaussian plume, a commonly used steady-state dispersion model, and the Gaussian puff, a time varying dispersion model that approximates a continuous release as a sum over many small "puffs". We compare model predictions to observations from a network of point-in-space continuous monitoring systems (CMS) collected during a series of controlled releases. Specifically, we use the Pearson correlation coefficient and mean absolute error (MAE) as metrics to assess the fit of the model predictions to the observed concentrations in terms of pattern and amplitude, respectively. The Gaussian puff outperforms the Gaussian plume using both metrics with average correlation coefficients of 0.38 and 0.31 and average MAEs of 0.70 and 0.74, respectively. We provide computationally efficient and scalable implementations of the Gaussian puff model. Compared to regulatory-grade, Gaussian puff-based models like CALPUFF, our implementations have higher spatial and temporal resolution and require only essential and

practically available meteorological information. These features enable near real-time methane mitigation applications on oil and gas sites and might be useful for near-field atmospheric transport modeling applications more broadly.

Keywords: Gaussian puff, Gaussian plume, methane emissions, oil and gas, continuous monitoring systems, code implementations

Synopsis: The Gaussian puff model outperforms the Gaussian plume model when characterizing near-field methane emissions on oil and gas sites.

1 Introduction

Reducing anthropogenic methane emissions will greatly increase the feasibility of the 1.5 degree temperature goal set at the 2015 Paris Climate Agreement.[1,2] The oil and gas sector provides a promising avenue for emissions reduction, as it accounts for 22% of global anthropogenic methane emissions[3,4] and 32% within the U.S.[5] Methane emissions from the oil and gas sector exhibit high temporal variability,[6,7] and infrequent, short-lived super emitter events represent a large portion of overall emissions.[8,9] Therefore, continuous monitoring systems (CMS) will likely play an increasing role in emissions monitoring, as they provide the near real-time measurements that are necessary for capturing both highly variable emissions and short-lived events.[10] Rather than directly measuring emission rates,[11] most CMS measure ambient methane concentrations at sensor locations (for point-in-space sensor networks),[12] integrated concentrations along an open-path (for scanning laser systems),[13] or 2D images of concentration enhancements (for IR cameras).[14]

For point-in-space sensors, inferring emission source information (i.e., location and rate) from concentration measurements can be framed as an inverse problem, which requires a forward model to simulate the transport of methane. Many inversions use the Gaussian plume atmospheric dispersion model,[15,16] which is easy to implement and computationally inexpensive but relies on a steady state assumption that is violated in practice during vari-

able wind conditions. Furthermore, averaging wind data over longer time periods to meet the steady state assumption often eliminates the pertinent high frequency elements of the signal. Looking at more complex models, large eddy simulations (LES) are sophisticated implementations of atmospheric transport that directly solve the governing equations using numerical computation techniques.[17,18] LES models are far more accurate than the Gaussian plume model but have a much higher computational cost and require special expertise to implement.

The Gaussian puff model strikes a balance between the Gaussian plume and LES models by relaxing the steady state assumption of the Gaussian plume model while allowing for time varying wind conditions and emission rates.[19] Therefore, the Gaussian puff model is better suited for simulating the transport of methane on oil and gas production sites in terms of accuracy compared to the Gaussian plume model and in terms of speed compared to the LES models. However, unlike the Gaussian plume model, the Gaussian puff model is still not widely used in this study area.[13,20,21] This could be due to a lack of evidence of its better performance or the unavailability of easy-to-use implementations. Existing implementations of the Gaussian puff model, e.g., CALPUFF[22] and SCIPUFF,[23] are designed for long-term and large-scale problems and require special expertise to implement. Also, they require detailed meteorological input data such as the vertical profiles of wind speed, wind direction, temperature, and pressure; data that are often unavailable in practice on oil and gas sites. In addition, the complexity of these models leads to high computational cost, which can hinder near real-time applications. Moreover, the spatial and temporal resolution of these models may not be sufficient for use on relatively small production oil and gas sites discussed in this work. To the best of our knowledge, the finest spatial resolution investigated in publicly available research using CALPUFF is about 100 meters,[24] which is too coarse for many methane mitigation applications on oil and gas production sites that can require a spatial resolution at the meter- or centimeter-scale. Finally, the software required to run these models are large and complex, and hence would be hard to modify and move to a

distributed computing environment for at-scale implementations.

This work addresses the aforementioned issues with three novel contributions. First, we provide the first (to our knowledge) direct comparison of the Gaussian plume and puff models in the context of modeling methane emissions on oil and gas sites, finding that the Gaussian puff outperforms the Gaussian plume across multiple measure-of-fit metrics. We limit our comparison to the Gaussian plume and puff models (and exclude LES models) because the Gaussian puff model is accurate enough on oil and gas production sites to be useful in practice[12] while also being lightweight enough to be used in near real-time without unreasonable computational burden. We perform the comparison using point-in-space CMS data collected during controlled releases at the Methane Emissions Technology Evaluation Center (METEC) in Fort Collins, Colorado. Second, we propose two practical criteria for setting a key parameter of the Gaussian puff model: the frequency at which puffs are generated; too few puffs result in an inaccurate simulation, while too many result in unnecessary computational expense. A previous criterion from Ludwig et al.[25] is theoretically sound, but nearly impossible to implement in practice, as it is a function of both location and wind speed and hence is different for each sensor and time stamp. Third, we make available two lightweight implementations of the Gaussian puff atmospheric dispersion model written in Python and R. These implementations are tailored for use on oil and gas sites by providing simulations with higher spatial and temporal resolution than regulatory-grade implementations like CALPUFF. Additionally, they require only essential and practically available meteorological information (horizontal wind speed and wind direction), and they can be easily scaled to distributed computing environments. Note that while we focus on applications related to oil and gas sites in this work, the proposed implementations can be easily adapted to different near-field atmospheric transport modeling applications.

2 Methods

In this section, we first describe the advection-diffusion partial differential equation (PDE) that governs the transport of methane in the atmosphere. We then introduce the Gaussian plume and Gaussian puff dispersion models as two analytical solutions to the advection-diffusion PDE under different sets of assumptions.

2.1 Advection-diffusion PDE

The transport of a non-reactive contaminant in the air (e.g., methane) is governed by the following advection-diffusion PDE:

$$\frac{\partial c}{\partial t} + \nabla \cdot (c\mathbf{u}) - \nabla \cdot (\mathbf{K}\nabla c) = s, \tag{1}$$

where the concentration of the contaminant, $c(x, y, z, t)$, is a function of time and space, $\mathbf{u}$ is the wind vector, $\mathbf{K}$ is a diagonal matrix whose elements are the diffusion coefficients k_x, k_y, and k_z in the x, y, and z direction, respectively, and $s(x, y, z, t)$ is the source term. The operator ∇ is the gradient of a scalar field and $\nabla\cdot$ is the divergence of a vector field. While our focus is from hereon exclusively on methane, the above PDE is broadly valid for non-reactive contaminants.

The expanded form of this PDE is given by

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + v\frac{\partial c}{\partial y} + w\frac{\partial c}{\partial z} + c(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}) - k_x\frac{\partial^2 c}{\partial x^2} - k_y\frac{\partial^2 c}{\partial y^2} - k_z\frac{\partial^2 c}{\partial z^2} = s, \tag{2}$$

where $\mathbf{u} = (u, v, w)$.

Complex numerical computational models based on LES can be used to solve this PDE without imposing any simplifying assumptions. Alternatively, at the sacrifice of accuracy, one can derive explicit solutions to the PDE by imposing a number of assumptions. Two such solutions are the Gaussian plume and Gaussian puff dispersion models. These solutions

are similar but have an important difference, namely the latter accounting for variable wind conditions and non-constant emission rates. This is important for near-field, short-time scale applications like methane emissions on oil and gas sites. In the following subsections, we list the assumptions made by the Gaussian plume and puff models and compare the form of the two models.

2.2 Gaussian plume model

The Gaussian plume model describes the contaminant concentration from a point source as a steady state plume whose dispersion is Gaussian in shape. The assumptions required to derive the Gaussian plume model from the advection-diffusion PDE are:

1. The source is a point source located at (x_0, y_0, H_0) with a constant emission rate q. That is, the source term as a function of space (x, y, z) and time t can be described as $s(x, y, z, t) = q\delta(x - x_0)\delta(y - y_0)\delta(z - H_0)$ where δ is the Dirac delta function.

2. The advection-diffusion process reaches steady state. That is, all quantities are constant in time including concentration c, wind vector $\mathbf{u}$, and diffusion coefficients k_x, k_y and k_z.

3. The wind speed and direction are constant across the domain. That is, $\mathbf{u} = (u, 0, 0)$ where $u \geq 0$ is constant and the downwind direction is aligned with the positive x-axis.

4. The diffusion coefficients are isotropic and are a function of downwind distance only. That is, $k_x = k_y = k_z = k(x)$.

5. The wind speed is assumed to be high enough that the diffusion effect in the downwind direction is much smaller than the advection effect and can be neglected. That is, $k_x \partial_x^2 c = 0$.

6. Topography and blockage are not considered. Hence, the model does not account for wakes generated by buildings and terrain and instead assumes free flow across the

domain.

Since we assume a constant wind field within the domain, we can set the coordinate system such that the source is located at $(0, 0, H_0)$ and the positive x-axis is aligned with the downwind direction. The domain of interest therefore becomes $\Omega = [0, \infty] \times [-\infty, \infty] \times [0, \infty]$. To solve the PDE, we also need appropriate boundary conditions. In the Gaussian plume model, two types of boundary conditions are implemented: (1) concentration vanishing at upwind locations and at infinity and (2) the vertical concentration flux does not penetrate the ground. All boundary conditions are summarized as:

$$c(0, y, z) = c(\infty, y, z) = c(x, \pm\infty, z) = c(x, y, \infty) = 0 \tag{3a}$$

$$k_z \frac{\partial c}{\partial z}\bigg|_{z=0} = 0. \tag{3b}$$

Combining the advection-diffusion governing equation (2), the corresponding boundary conditions (3), and the assumptions listed above, we can obtain the Gaussian plume solution, given by

$$c(x, y, z) = \frac{q}{2\pi u \sigma_y \sigma_z} \exp\left(-\frac{y^2}{2\sigma_y^2}\right) \left[\exp\left(-\frac{(z - H_0)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z + H_0)^2}{2\sigma_z^2}\right)\right], \tag{4}$$

where $\sigma_y = \sigma_y(x)$ and $\sigma_z = \sigma_z(x)$ are dispersion parameters that control the width of the plume in the y and z directions, respectively, and are functions of downwind distance x only. The dispersion parameters are typically computed using the empirical rules given below:[26,27]

$$\begin{aligned} \sigma_z &= a x^b \\ \sigma_y &= 465.11628 x \tan\Theta \\ \Theta &= 0.017453293[c - d\log(x)], \end{aligned} \tag{5}$$

where the units of σ_y and σ_z are [m] and the units of x are [km]. The a, b, c, d parameter

values in (5) are a function of the Pasquill atmospheric stability class and can be obtained from lookup tables.[28]

In summary, by assuming steady state conditions, we can use the Gaussian plume model (4) to compute the concentration at any location (x, y, z) within the domain given a source location $(0, 0, H_0)$, constant emission rate q, and constant wind speed u.

2.3 Gaussian puff model

The Gaussian plume model provides a simple and effective way to compute concentrations under steady state conditions. However, the steady state assumption is often invalid in practice, and hence the Gaussian puff model was proposed to better accommodate time varying wind and emission rates.[29,30] The Gaussian puff model approximates a continuous emission as a series of discrete puffs emitted successively from a point source, with the overall predicted concentration being the sum of the concentration contribution of each individual puff.

The Gaussian puff model shares the same advection-diffusion governing equation and boundary conditions as the Gaussian plume model, but it relaxes the steady state assumption (the second assumption in Section 2.2), making the concentration prediction a function of both space and time, i.e., $c = c(x, y, z, t)$. In addition, the constant emission source in the Gaussian plume model is replaced with a separate, instantaneous source for each puff. The concentration contribution of a single puff p is given by

$$c_p(x, y, z, t) = \frac{Q}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2 + y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H_0)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H_0)^2}{2\sigma_z^2}\right)\right], \tag{6}$$

where Q is the mass of methane contained in puff p. The overall concentration prediction at the sensor locations (x, y, z) is taken to be the sum of the concentration contribution of all effective puffs, given by

$$c(x, y, z, t) = \sum_{p \in \mathcal{S}_t} c_p(x, y, z, t), \tag{7}$$

where $\mathcal{S}_t$ is the set of effective puffs at time t. $\mathcal{S}_t$ can be defined in different ways, all of which discard puffs that have been transported far from the domain of interest and hence would contribute very little to the overall concentration prediction at the evaluation locations. One option is to discard puffs after a fixed lifetime that is set based on domain size and average wind speed. Another option is to discard puffs once they reach a given distance from the domain. We use the former option to generate the simulation results discussed here.

Note that the performance of the Gaussian puff model depends on how often puffs are created, which we call the puff simulation frequency. We discuss puff simulation frequency in Section 3.2 and provide guidance on how to set this parameter.

Because it can accommodate time varying emission rates and wind conditions, simulation predictions from the Gaussian puff model are higher fidelity than from the Gaussian plume model. We compare output from the Gaussian plume and puff models in Section 4.2. Furthermore, the Gaussian puff model requires notably less computational resources than an LES model. Therefore, the Gaussian puff model is well suited for modeling the transport of methane on relatively small oil and gas production sites for near real-time use cases.

3 Implementation of the Gaussian puff model

In this section, we describe our implementation of the Gaussian puff model and propose simple guidelines for choosing an appropriate puff simulation frequency.

3.1 Gaussian puff algorithm

We implement the Gaussian puff model in two programming languages: Python and R. A high level description of the Python algorithm is given here. For brevity, a similar description of the R algorithm and pseudo code for both implementations can be found in Sections S1 and S2 in the supporting information (SI) file. The Python implementation proceeds as follows:

1. Discretize a continuous emission by a series of puffs according to the specified puff simulation frequency.

2. For each puff, do the following:

   (a) Use the wind speed and direction from the time of puff creation to determine that puff's movement. Note that to be consistent with the derivation of the Gaussian puff model, we assume that wind speed and direction are constant during each puff's lifetime but can vary from one puff to the next. This assumption is reasonable in practice, as each puff persists for only a few seconds to minutes on a typical oil and gas production site and new puffs are created frequently.

   (b) Rotate the coordinate system so that the positive x-axis is inline with the downwind direction. Compute the dispersion parameters σ_y and σ_z based on the wind speed.

   (c) Compute the concentration contribution of the given puff at the sensor locations using Equation 6. Continue to compute the concentration contribution of the given puff for each subsequent time stamp until the puff is discarded when an early stop criterion is satisfied (either based on puff lifetime or distance from the area of interest).

3. Sum the concentration contribution of all puffs to obtain the total concentration prediction at each sensor location, see Equation 7.

Note that we also implement a number of speed up improvements and memory optimizations in both languages. For example, in Python, we use the broadcast functionality from TensorFlow[31] to convert Equation 6 from scalar mode to tensor operations and hence eliminate inefficient nested loops. In R, to reduce memory requirements, we simulate over smaller time intervals in parallel and then stitch together the concentration output from each interval. In addition, both implementations take advantage of parallelization and hence can be

scaled on a distributed computing environment. More details on the speed up improvements can be found in Sections S3 and S4 in the SI.

3.2 Investigation of puff simulation frequency

The accuracy and computational cost of the Gaussian puff model largely depends on the frequency at which puffs are created. A higher puff simulation frequency results in a better approximation of a continuous release but comes at a higher computational cost. Therefore, the puff simulation frequency used in practice must strike a balance between acceptable simulation accuracy and the resulting computational expense. In the following discussion, we interchangeably use "puff simulation frequency" and the "time interval between two successive puffs", denoted by puff_dt. The two terms refer to the same parameter with the former in the frequency domain and the latter in the time domain.

We now use a toy example to illustrate the importance of simulation frequency. We run the Gaussian puff model in the 2-D domain [0, 40] m $\times$ [-15, 15] m and place a constant emission source of 1 kg/hr at the origin with constant wind blowing at 1 m/s along the positive x-axis. We place three sensors at coordinates $(10, 0), (20, 0), (30, 0)$ which are 10, 20, and 30 meters downwind of the source, respectively. The simulation is run for 120 seconds. Figure 1 shows the output from the Gaussian puff model at time stamps [30, 60, 90, 120] seconds using puff_dt values of [30, 20, 10, 1, 0.1] seconds to set the time between subsequent puffs. The four leftmost columns show the methane concentrations over space, and the rightmost column shows the time series of predicted methane concentrations at the three sensor locations. Each row in the figure shows the simulation predictions corresponding to a different puff_dt value.

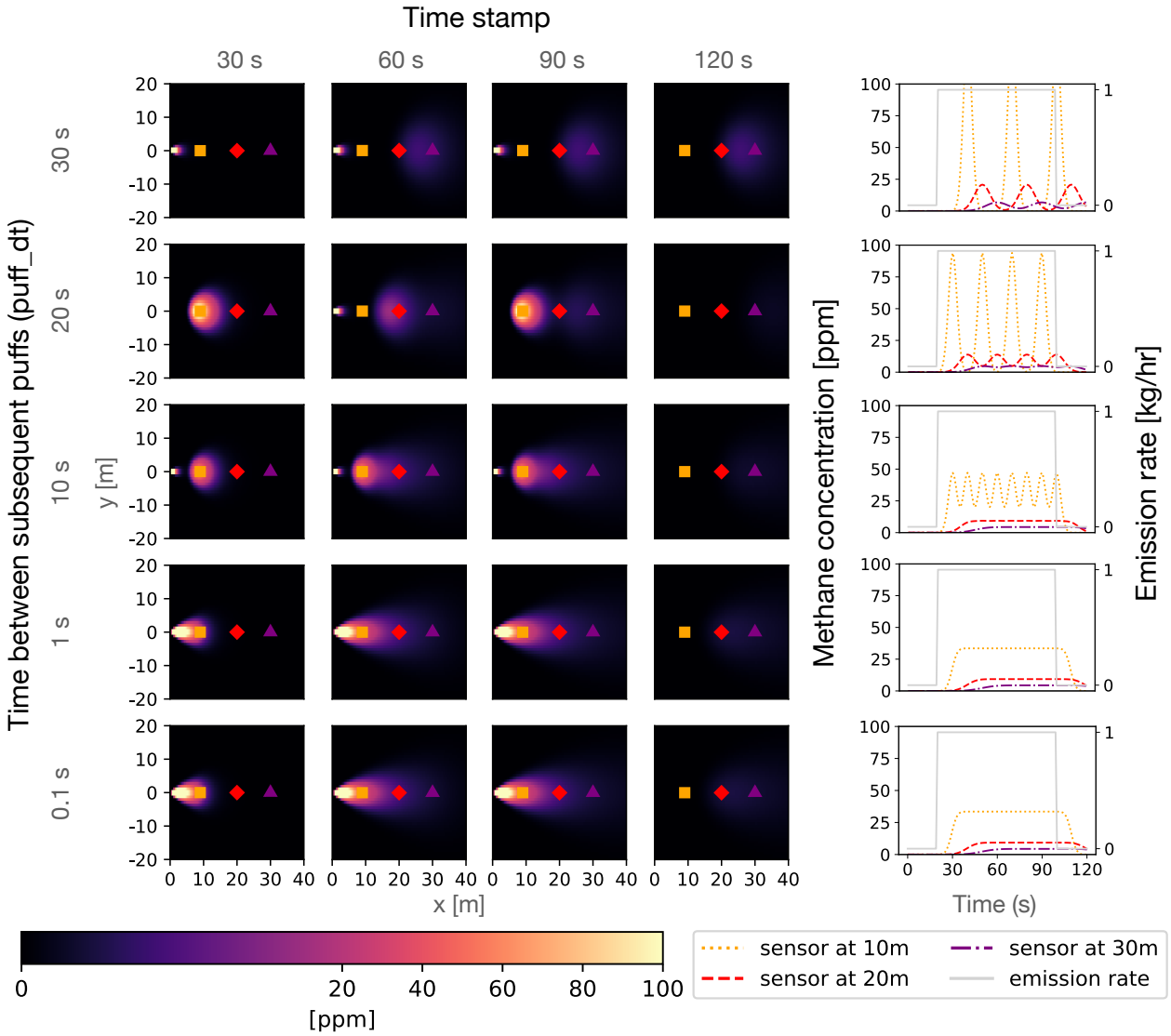


Figure 1: Toy example to demonstrate the importance of the puff simulation frequency. Rows correspond to different simulation frequencies. First four columns show simulation predictions over space at different time stamps in the simulation. The orange square, red diamond, and purple triangle show the sensor locations. Rightmost column shows a time series of simulation predictions at the three sensor locations (dotted orange, dashed red, and dash-dotted purple lines) and of the emission rate used to generate the simulations (solid gray line).

Since the emission rate and wind vector are constant in this toy example, the concentrations at each sensor location should reach a steady state after a short period of time assuming an infinitely small puff_dt value is used. This is because integrating the Gaussian puff solution in time yields the Gaussian plume solution by definition, given a constant

emission rate. However, steady state concentrations only occur for the smallest two puff_dt values (1 and 0.1 s) in Figure 1. The higher puff_dt values result in an oscillatory pattern in the concentration time series that is a result of gaps between the puffs. For example, in the second row (puff_dt = 20 s), the orange square sensor is at the center of a puff for some time stamps (30 s and 90 s) and between puffs for other time stamps (60 s). In the last two rows (puff_dt = 1 and 0.1 s), however, the orange square sensor records steady state concentrations once the puffs have fully passed over the sensor, as the gaps between puffs at these higher puff frequencies are very small.

This toy example naturally leads to the question: how should we select an appropriate puff simulation frequency that is neither too low (to avoid spurious artifacts) nor too high (to avoid unnecessary computational cost)? Ludwig et al.[25] suggests that the distance between two adjacent puffs should be less than $2\sigma_y$ (see the definition of σ_y in Equation 5). However, this threshold is hard to implement in practice, as σ_y is a function of location and wind speed and hence is different for each sensor location and time stamp. Here we propose two criteria that are easier to implement. The first depends on the time resolution of the CMS observations. To avoid sacrificing information contained in these observations, the puff_dt should be no larger than the time resolution of the observations, which leads to

$$\text{puff\_dt} \leq \text{cms\_dt}, \tag{8}$$

where cms_dt is the time resolution of the CMS observations. Experimentation with both simulated and real data has revealed that a puff simulation frequency much below the time resolution of the CMS data does not notably improve the simulation results.

The second criterion accounts for wind speed and the geometry of the domain. To avoid artifacts due to the discrete nature of the puffs, we want to prevent each sensor from being at the center of a puff or in the gap between two adjacent puffs. This is guaranteed if at least two puffs are generated between the source and the closest downwind sensor for all

wind speeds, which leads to

$$\text{puff\_dt} \leq d_{\min}/2u_{\max}, \tag{9}$$

where $d_{\min}$ is the distance between the source and the closest sensor and $u_{\max}$ is the maximum observed wind speed. Note that this criterion is very conservative because the maximum observed wind speed may be an outlier and the wind may not blow from the source to the closest sensor when the maximum wind speed occurs. Therefore, this criterion can serve as a lower bound for puff_dt. If this criterion is hard to achieve given the available computational resources, the 90^{th} percentile of the wind speed observations can be used instead of the max. Ideally, the puff_dt value would be allowed to change over time as the average wind speed shifts, e.g., from night to day, but we leave the implementation of this strategy to future work.

In practice, the second criterion will likely result in lower puff_dt values (i.e., higher puff frequencies) than the first. Therefore, the first criterion can be treated as an upper bound on puff_dt and the second criterion can be treated as a lower bound. Smaller puff_dt values result in a more accurate simulation, hence the smallest value between the two bounds that can be accommodated by the available computational resources should be selected.

4 Results

4.1 Data description and preprocessing

We evaluate the Gaussian puff model using concentration observations from a point-in-space CMS sensor network at the Methane Emissions Technology Evaluation Center (METEC) in Fort Collins, Colorado during the Advancing Development of Emissions Detection (ADED) research program (see Zimmerle[32] for details). The layout of the METEC site and the configuration of the ADED experiment are shown in Figure 2. METEC is a testing center that resembles an oil and gas production site and performs controlled methane releases from

multiple pieces of equipment. Data were collected from February 1 to May 16, 2022, by Project Canary[33] CMS sensors. There are five main equipment groups on the METEC site from which controlled releases can originate: West Wellhead, West Separator, Tanks, East Separator, and East Wellhead. Eight sensors were deployed around the perimeter of the METEC site, three of which (E, NW, and SW) were equipped with an anemometer that records wind speed and wind direction. Methane concentration, wind speed, and wind direction measurements are taken every second and then averaged and reported every minute. There were a total of 343 emission events during the ADED research program varying from 17 minutes to 8.25 hours in duration. The source location, emission rate, and the start and end time of each emission event are known, making it possible to compare the Gaussian plume and puff models simulated with the true emission rate to the CMS observations. Note that no information on emission timing, location, and rate is known in practice, but we use this information here to assess the performance of the Gaussian plume and Gaussian puff models by comparing the simulations to the CMS observations.

Because we assume a constant wind field over space (the third assumption described in Section 2.2), we take the minute-by-minute median of the wind data recorded by the three anemometers at each time stamp to generate a single wind speed and direction time series for the entire site. To simulate using the Gaussian puff model with a puff_dt value less than 60 seconds, we downsample the CMS data using linear interpolation. For emission events with multiple sources, we simulate for each source separately and then add the resulting concentrations from each simulation, which relies on the assumption that methane concentrations are additive in Gaussian plume and puff models. Note that neither the Gaussian plume nor puff models account for the background methane concentrations that are always present in the atmosphere (and hence in the CMS observations). To address this point, we subtract the minimum concentration value from each sensor from all other concentration observations recorded by that sensor. Note that more nuanced methods for background correction exist[12] but are not addressed in this paper.

The Gaussian plume and Gaussian puff simulations discussed in this paper were run on the high performance computing (HPC) system from the National Center for Atmospheric Research (NCAR). We used 36 CPUs on a single node and 109GB CPU memory. We simulate each emission event in parallel. The wall time for each simulation using our Python implementation is: 22 seconds for Gaussian plume, 5 minutes and 32 seconds for Gaussian puff with puff_dt = 60 s, and 4 hours, 40 minutes and 40 seconds for Gaussian puff with puff_dt = 1 s. The cumulative emission event duration for the entire ADED experiment is about 44.5 days, and hence the difference in computational cost between the Gaussian plume and puff will be much smaller for a single emission event.

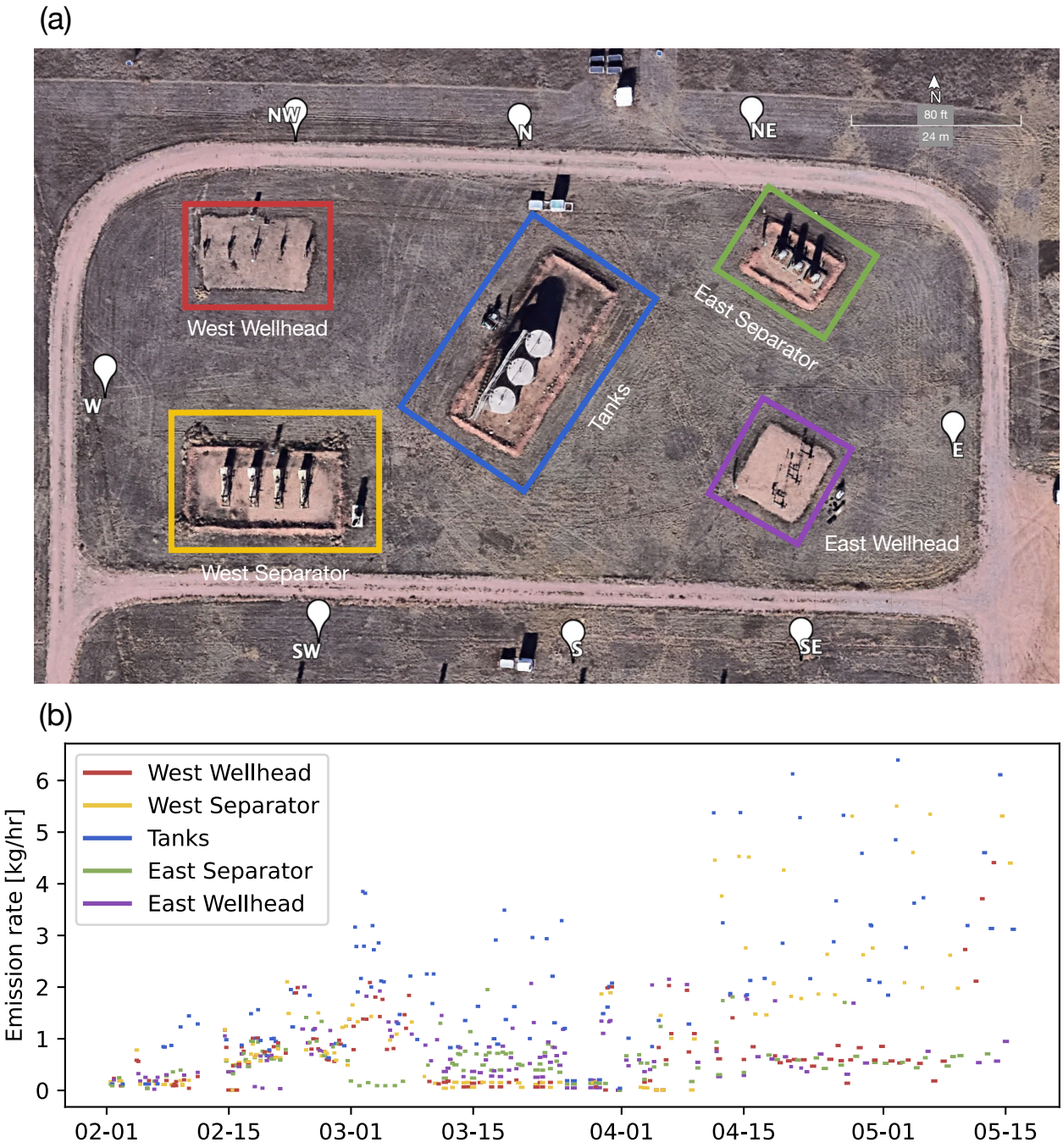


Figure 2: Panel (a) shows the satellite map of METEC site with five main equipment groups highlighted by rectangles in different colors and eight CMS sensors denoted by white markers. Panel (b) shows the rate [kg/hr] for all emission events within the METEC ADED experiment with colors denoting the source locations.

4.2 Comparison of Gaussian plume and puff simulation predictions

We now compare the simulated concentrations from the Gaussian plume and Gaussian puff models. For the Gaussian plume, we divide each emission event into multiple non-overlapping 5-minute intervals, in which we assume the transport of methane reaches steady state and hence use of the Gaussian plume model is valid. Within each interval, we use the mean emission rate, mean wind speed, and circular mean of the wind direction as inputs to the model (see Equation 4). For the Gaussian puff model, we use the algorithm described in Section 3.1 and use a puff_dt of 1 second. As an example to demonstrate the difference between these two models, Figure 3 shows the observations and simulation output from both the plume and puff models at the NW sensor from an emission event that occurred on April 18, 2022 between 12:47 - 15:17 (MST) with two emission sources: one at West Wellhead with emission rate 0.67 kg/hr and another one at East Wellhead with emission rate 1.73 kg/hr. The Gaussian puff model creates more detailed concentration predictions, as it is able to accommodate time varying wind conditions. The Gaussian plume model, on the other hand, is characterized by periods of constant concentration predictions corresponding to the 5-minute intervals over which the steady state assumption is made. In addition, the Gaussian puff simulation is closer to the observations in terms of amplitude compared to the Gaussian plume simulation.

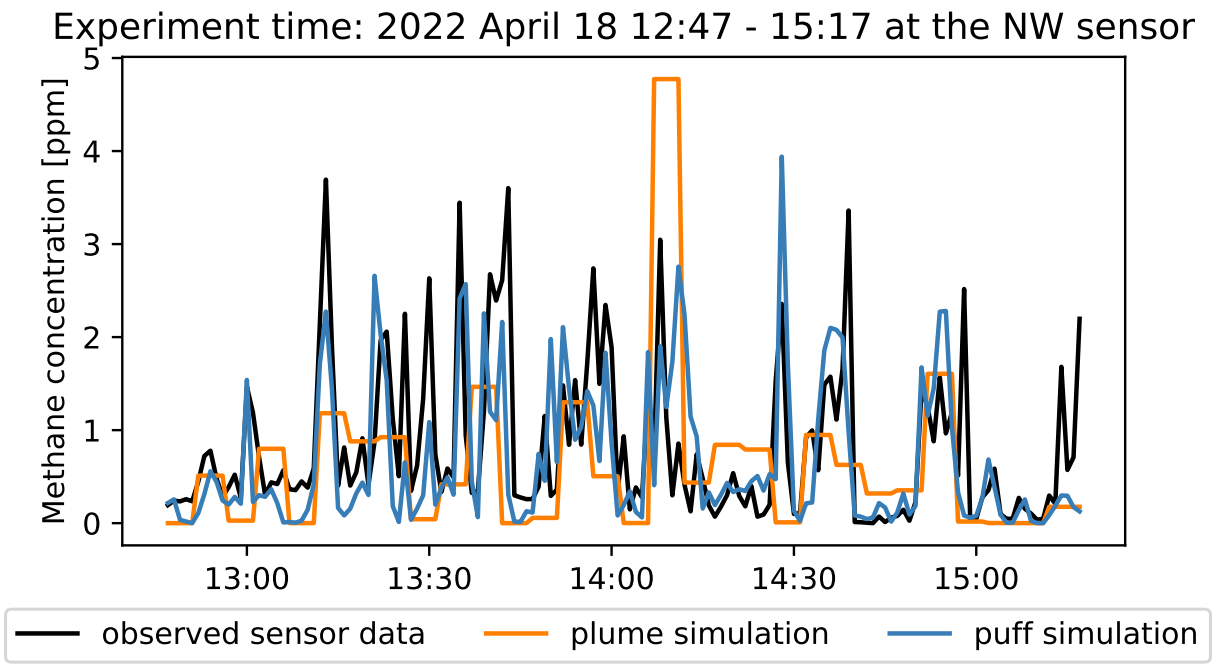


Figure 3: A comparison of simulation predictions from the Gaussian plume (orange line) and puff (blue line) models. The selected emission event had two sources: West Wellhead with an emission rate of 0.67 kg/hr and East Wellhead with an emission rate of 1.73 kg/hr. CMS observations are shown as a black line.

For a more comprehensive comparison, we now consider all emission events that occurred during the METEC ADED experiment. We assess the accuracy of both models by comparing their output to the CMS observations using the Pearson correlation coefficient, which measures similarity in patterns without accounting for amplitude, and the mean absolute error (MAE), which measures the averaged absolute difference in amplitude between the simulation and the observation. Specifically, for each emission event, we compute both metrics for each sensor and then average the metric values across all 8 sensors. Figure 4 summarizes model accuracy throughout the experiment, with each cross representing a separate emission event. In Subfigure (a), we see that in general, the Gaussian puff simulations are more correlated to the observations compared to the Gaussian plume simulations. Note that for many emission events either the observations or predictions for a given sensor are all zero, which yields an undefined correlation coefficient. To account for this, we add the

same Gaussian noise to both the observations and predictions. The noise is centered at 0 ppm with a standard deviation of 0.01 ppm, which is small enough to have essentially no impact on the correlation coefficient for emission events that have nonzero observations and predictions and results in a correlation coefficient of 1 when the observations and predictions are both zero. In Subfigures (b) and (c), we can see that, in general, the Gaussian puff simulations are closer to the observations in terms of MAE. The difference in MAE between the two models is smaller than the difference in correlation coefficient. This is likely because of the inadequacy of both models (e.g., both ignore diffusion in calm wind conditions), and hence the difference between either model and the actual observations dominates over the difference between the two models. We also compute the overall correlation coefficient and MAE by taking the average across all the emission events. The overall correlation coefficient of the Gaussian plume and puff models is 0.31 and 0.38, respectively. The overall MAE of Gaussian plume and puff models is 0.74 and 0.70, respectively. These metrics reveal that the Gaussian puff model outperforms the Gaussian plume model.

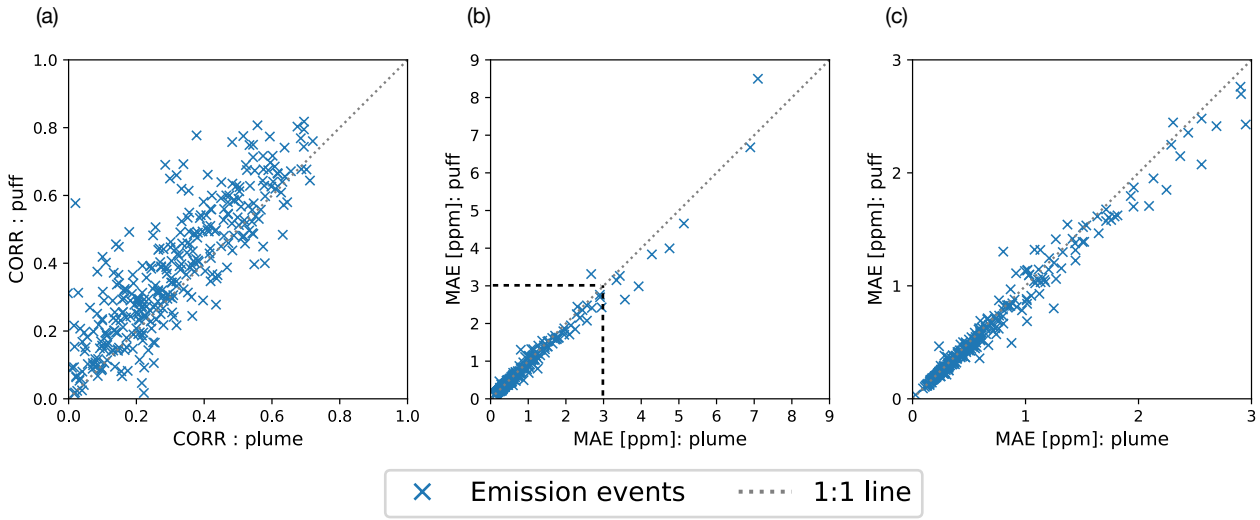


Figure 4: Comparison of the Gaussian plume and puff models across the entire METEC ADED experiment. Panel (a): Pearson correlation coefficient between observations and Gaussian plume predictions (x axis) and Gaussian puff predictions (y axis) for each emission event. The overall correlation coefficient is 0.31 and 0.38 for the Gaussian plume and puff model, respectively. Panel (b): Same as (a) but using mean absolute error (MAE) as the metric. The overall MAE is 0.74 and 0.70 for the Gaussian plume and puff model, respectively. Dashed box delineates the emission events that are zoomed in on in (c). Panel (c): Zoom in on (b) to show detail.

4.3 Comparison of Gaussian puff predictions using different simulation frequencies

Here we compare Gaussian puff simulation predictions generated using different puff_dt values. We again use data from the METEC ADED experiment to conduct the comparison. Recall that the temporal resolution of the CMS observations is 1 minute, i.e., cms_dt = 1 minute, and the closest source-sensor pair on the METEC site has a distance of about 15 m, i.e., $d_{\min} = 15$ m. Since the maximum wind speed throughout the ADED experiment is 18.12 m/s, which is a significant outlier, we choose to use the 0.9 quantile value of 5.39 m/s to determine puff_dt in Equation 9. Hence, the recommended puff_dt values given by the two criteria are 60 s and 1.39 s, respectively. Based on these values, we use puff_dt values of 60 s and 1 s to perform the comparison in this section. Similar to Section 4.2, we compare the puff and plume models using the Pearson correlation coefficient and mean absolute error

(MAE). The two puff_dt values produce very similar simulation results in terms of both metrics, except for a few cases where 1 s results in notably higher correlation than 60 s. The overall correlation coefficients are 0.35 and 0.38 for puff_dt = 60 s and 1 s, respectively. The overall MAE are 0.72 and 0.70 for puff_dt = 60 s and 1 s, respectively.

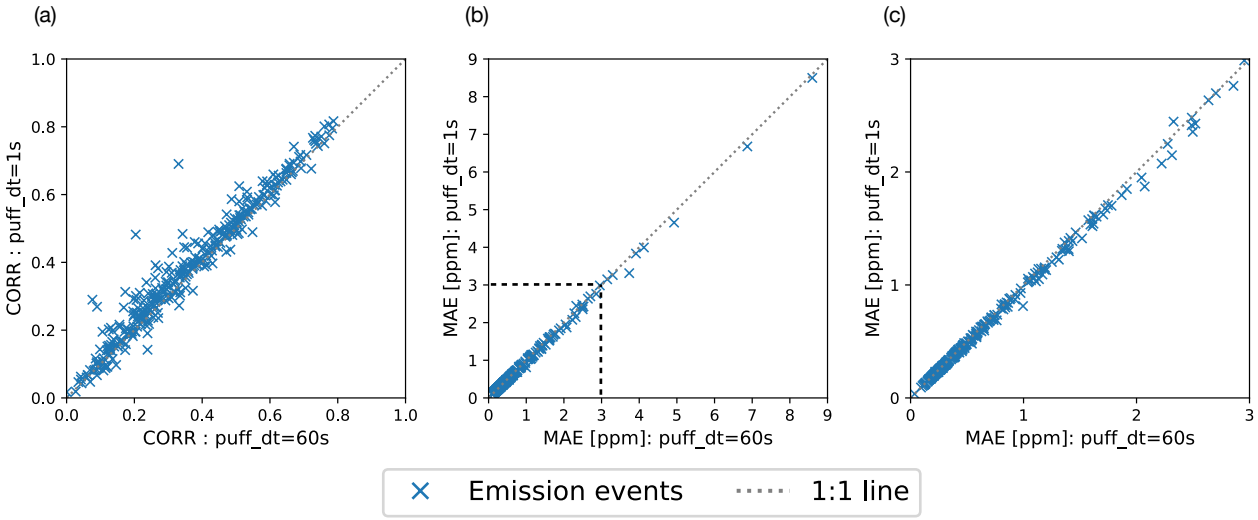


Figure 5: Comparison of two different puff simulation frequencies (1 s and 60 s) across the entire METEC ADED experiment. Panel (a): Pearson correlation coefficient between observations and the Gaussian puff predictions with puff_dt = 60 s (x axis) and puff_dt = 1 s (y axis) for each emission event. The overall correlation coefficient is 0.35 and 0.38 for puff_dt values of 60 s and 1 s, respectively. Panel (b): Same as (a) but using mean absolute error (MAE) as the metric. The overall MAE is 0.72 and 0.70 for puff_dt = 60 s and 1 s, respectively. Dashed box delineates the emission events that are zoomed in on in (c). Panel (c): Zoom in on (b) to show detail.

5 Discussion

In this paper, we compare the Gaussian plume and puff atmospheric dispersion models with a focus on methane emissions on oil and gas production sites. We find that the Gaussian puff model outperforms the Gaussian plume model in terms of correlation coefficient and MAE between CMS observations and model predictions. Furthermore, we provide guidelines for setting the frequency at which puffs are generated in the Gaussian puff model, which affects the simulation accuracy and computational cost. Finally, we provide open-source implementations of the Gaussian puff model in both Python and R that are tailored for use

in near real-time methane mitigation applications on oil and gas production sites. While our focus is specifically on the applications on oil and gas sites in this work, the proposed implementations can be easily adapted for use in various near-field scenarios.

The time varying nature of the Gaussian puff model makes it especially advantageous compared to the Gaussian plume model in the following scenarios.

- CMS concentration and wind data are available at a high temporal resolution. The Gaussian puff model is able to exploit these high resolution data, resulting in more accurate concentration predictions. For use with the Gaussian plume model the high-resolution data would need to be averaged over longer time periods to meet the steady state assumption, which eliminates the higher frequency component of the signal.

- Near real-time methane mitigation applications such as emission detection, localization, and quantification are desired. In this case, using an extended averaging window for the Gaussian plume model may introduce too long of a lag between observations and detection, localization, and quantification results.

- Emission rates are not constant. This scenario violates the steady state assumption in the Gaussian plume model but is easily accommodated by the Gaussian puff model.

Our implementations of the Gaussian puff model are better suited for near real-time applications on oil and gas sites than regulatory-grade models such as CALPUFF for the following reasons:

- They are lightweight and require less computational cost, which facilitates a quick response to methane emission on oil and gas sites.

- They are specifically designed for the scale of a typical oil and gas production site, as they operate at a higher spatial and temporal resolution than CALPUFF (about 100 meters).

- They require limited knowledge of atmospheric transport modeling and require only source and sensor locations and horizontal wind speed and direction as input data.
- They are developed in Python and R with built-in functionality for parallel computing, which can be easily implemented at scale.

Future work will investigate the use of the Gaussian puff model on more complex sites (e.g., compressor stations) where large buildings can disrupt advective transport. Note that the Gaussian puff model does not account for building wake effects, topography, and complex boundary conditions. Therefore, a more sophisticated forward model may be necessary for accurate concentration predictions on more complex sites that have large obstructions (e.g., buildings), are in locations with highly variable topography, or have complex boundary conditions due to, e.g., a steep incline on one or more sides of the site. It is also worth reiterating that the Gaussian puff model does not account for diffusion along the wind direction (see the fifth assumption described in Section 2.2). Therefore, the simulated concentrations from both the Gaussian plume and puff models will be zero at any upwind locations, which is not true when the wind speed is low and upwind sensors are located close to the emission source. We think that this issue contributes to the mismatch between observations and simulation output during the experiments studied in this paper and will further investigate this topic in future work. Despite these inadequacies, however, we have found that the Gaussian puff model is accurate enough on relatively simple production-style sites for use in methane emission detection, localization, and quantification algorithms.[12]

Acknowledgement

The authors thank Project Canary for sharing their raw sensor data. The authors also acknowledge high-performance computing support from Cheyenne (doi:10.5065/D6RX99HX) provided by NCAR's Computational and Information Systems Laboratory, sponsored by the National Science Foundation.

Supporting Information Available

The following files are available free of charge.

- Supporting information: additional details on the implementation of the Gaussian puff model in Python and R.

References

(1) Collins, W. J.; Webber, C. P.; Cox, P. M.; Huntingford, C.; Lowe, J.; Sitch, S.; Chadburn, S. E.; Comyn-Platt, E.; Harper, A. B.; Hayman, G.; Powell, T. Increased importance of methane reduction for a 1.5 degree target. *Environmental Research Letters* **2018**, *13*, 054003.

(2) Schleussner, C. F.; Rogelj, J.; Schaeffer, M.; Lissner, T.; Licker, R.; Fischer, E. M.; Knutti, R.; Levermann, A.; Frieler, K.; Hare, W. Science and policy characteristics of the Paris Agreement temperature goal. *Nature Climate Change 2016 6:9* **2016**, *6*, 827–835.

(3) O'Rourke, P. R.; Smith, S. J.; Mott, A.; Ahsan, H.; McDuffie, E. E.; Crippa, M.; Klimont, Z.; McDonald, B.; Wang, S.; Nicholson, M. B.; Feng, L.; Hoesly, R. M. CEDS v-2021-02-05 Emission Data 1975-2019. `http://doi.org/10.5281/zenodo.4509372`, 2021; Zenodo.

(4) Crippa, M.; Guizzardi, D.; Muntean, M.; Schaaf, E.; Lo Vullo, E.; Solazzo, E.; Monforti-Ferrario, F.; Olivier, J.; Vignati, E. EDGAR v6.0 Greenhouse Gas Emissions. `http://data.europa.eu/89h/97a67d67-c62e-4826-b873-9d972c4f670b`, 2021; European Commission, Joint Research Centre (JRC).

(5) EPA, *Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990-2020*; 2022.

(6) Allen, D. T.; Cardoso-Saldaña, F. J.; Kimura, Y. Variability in Spatially and Temporally Resolved Emissions and Hydrocarbon Source Fingerprints for Oil and Gas Sources in Shale Gas Production Regions. *Environmental Science and Technology* **2017**, *51*, 12016–12026.

(7) Wang, J. L.; Daniels, W. S.; Hammerling, D. M.; Harrison, M.; Burmaster, K.; George, F. C.; Ravikumar, A. P. Multiscale Methane Measurements at Oil and Gas Facilities Reveal Necessary Frameworks for Improved Emissions Accounting. *Environmental Science & Technology* **2022**, *56*, 14743–14752.

(8) Brandt, A. R.; Heath, G. A.; Cooley, D. Methane Leaks from Natural Gas Systems Follow Extreme Distributions. *Environmental Science and Technology* **2016**, *50*, 12512–12520.

(9) Cusworth, D. H.; Duren, R. M.; Thorpe, A. K.; Olson-Duvall, W.; Heckler, J.; Chapman, J. W.; Eastwood, M. L.; Helmlinger, M. C.; Green, R. O.; Asner, G. P.; Dennison, P. E.; Miller, C. E. Intermittency of Large Methane Emitters in the Permian Basin. *Environmental Science and Technology Letters* **2021**, *8*, 567–573.

(10) Bell, C.; Ilonze, C.; Duggan, A.; Zimmerle, D. Performance of Continuous Emission Monitoring Solutions under a Single-Blind Controlled Testing Protocol. *Environmental Science and Technology* **2023**, *57*, 5794–5805.

(11) Riddick, S. N.; Ancona, R.; Mbua, M.; Bell, C. S.; Duggan, A.; Vaughn, T. L.; Bennett, K.; Zimmerle, D. J. A quantitative comparison of methods used to measure smaller methane emissions typically observed from superannuated oil and gas infrastructure. *Atmospheric Measurement Techniques* **2022**, *15*, 6285–6296.

(12) Daniels, W.; Jia, M.; Hammerling, D. Methane emission detection, localization, and quantification using continuous point-sensors on oil and gas facilities. *ChemRxiv* **2022**,

(13) Alden, C. B.; Coburn, S. C.; Wright, R. J.; Baumann, E.; Cossel, K.; Perez, E.; Hoenig, E.; Prasad, K.; Coddington, I.; Rieker, G. B. Single-blind quantification of natural gas leaks from 1 km distance using frequency combs. *Environmental science & technology* **2019**, *53*, 2908–2917.

(14) Abdel-Moati, H.; Morris, J.; Zeng, Y.; Kangas, P.; McGregor, D. New optical gas imaging technology for quantifying fugitive emission rates. International Petroleum Technology Conference. 2015.

(15) Sharan, M.; Issartel, J. P.; Singh, S. K.; Kumar, P. An inversion technique for the retrieval of single-point emissions from atmospheric concentration measurements. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **2009**, *465*, 2069–2088.

(16) Kumar, P. et al. Near-field atmospheric inversions for the localization and quantification of controlled methane releases using stationary and mobile measurements. *Quarterly Journal of the Royal Meteorological Society* **2022**, *148*, 1886–1912.

(17) Keats, A.; Yee, E.; Lien, F. S. Bayesian inference for source determination with applications to a complex urban environment. *Atmospheric Environment* **2007**, *41*, 465–479.

(18) Travis, B.; Dubey, M.; Sauer, J. Neural networks to locate and quantify fugitive natural gas leaks for a MIR detection system. *Atmospheric Environment: X* **2020**, *8*, 100092.

(19) Stockie, J. M. The mathematics of atmospheric dispersion modeling. *Siam Review* **2011**, *53*, 349–372.

(20) Riddick, S. N.; Connors, S.; Robinson, A. D.; Manning, A. J.; Jones, P. S.; Lowry, D.; Nisbet, E.; Skelton, R. L.; Allen, G.; Pitt, J., et al. Estimating the size of a methane emission point source at different scales: from local to landscape. *Atmospheric Chemistry and Physics* **2017**, *17*, 7839–7851.

(21) Kumar, P.; Broquet, G.; Caldow, C.; Laurent, O.; Gichuki, S.; Cropley, F.; Yver-Kwok, C.; Fontanier, B.; Lauvaux, T.; Ramonet, M., et al. Near-field atmospheric inversions for the localization and quantification of controlled methane releases using stationary and mobile measurements. *Quarterly Journal of the Royal Meteorological Society* **2022**, *148*, 1886–1912.

(22) Scire, J. S.; Strimaitis, D. G.; Yamartino, R. J., et al. A user's guide for the CALPUFF dispersion model. *Earth Tech, Inc* **2000**, *521*, 1–521.

(23) Sykes, R.; Parker, S.; Henn, D.; Gabruk, R. SCIPUFF—A generalized dispersion model. *Air Pollution Modeling and Its Application XI* **1996**, 425–432.

(24) Rzeszutek, M. Parameterization and evaluation of the CALMET/CALPUFF model system in near-field and complex terrain-Terrain data, grid resolution and terrain adjustment method. *Science of the Total Environment* **2019**, *689*, 31–46.

(25) Ludwig, F.; Gasiorek, L.; Ruff, R. Simplification of a Gaussian puff model for real-time minicomputer use. *Atmospheric Environment (1967)* **1977**, *11*, 431–436.

(26) Pasquill, F. The Estimation of the Dispersion of Windborne Material. *Meteorological Magazine* **1961**, *90*, 33–49.

(27) Turner, B. D. *Workbook of atmospheric dispersion estimates*; 1970.

(28) EPA, *Workbook for plume visual impact screening and analysis (revised)*; 1992.

(29) Llewelyn, R. P. An analytical model for the transport, dispersion and elimination of air pollutants emitted from a point source. *Atmospheric Environment (1967)* **1983**, *17*, 249–256.

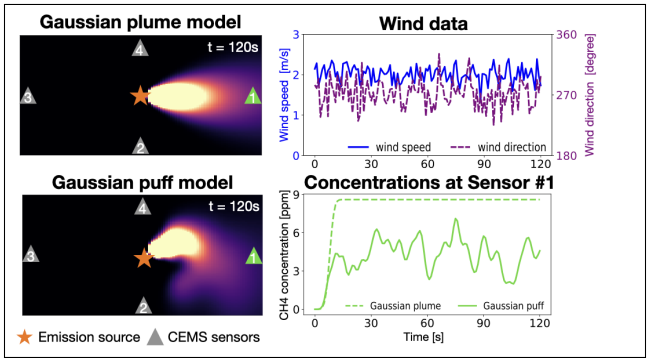
(30) Okamoto, S.; Ohnishi, H.; Yamada, T.; Mikami, T.; Momose, S.; Shinji, H.; Itohiya, T. A model for simulating atmospheric dispersion in low-wind conditions. *International journal of environment and pollution* **2001**, *16*, 69–79.

(31) Abadi, M. et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015; `https://www.tensorflow.org/`, Software available from tensorflow.org.

(32) Zimmerle, D. *METEC Controlled Test Protocol: Continuous Monitoring Emission Detection And Quantification*; 2020.

(33) Project Canary. `https://www.projectcanary.com/`, 2022.

TOC Graphic


Gaussian plume model
t = 120s
Gaussian puff model
t = 120s
Emission source
CEMS sensors
Wind data
Wind speed [m/s]
Wind direction [degree]
wind speed
wind direction
Concentrations at Sensor #1
CH4 concentration [ppm]
Gaussian plume
Gaussian puff
Time [s]