Environmental Data Science (2026), 5: e2, 1–29
doi:10.1017/eds.2026.10029

METHODS PAPER

Actively inferring methane sources with drones

Alouette van Hove, Kristoffer Aalstad and Norbert Pirk

Department of Geosciences, University of Oslo, Oslo, Norway
Corresponding author: Alouette van Hove; Email: a.van.hove@geo.uio.no

Received: 17 September 2025; **Revised:** 20 November 2025; **Accepted:** 05 January 2026

Keywords: adaptive informative path planning; Bayesian inference; drones; methane monitoring; reinforcement learning

Abstract

This study presents a framework that combines Bayesian inference with reinforcement learning to guide drone-based sampling for methane source estimation. Synthetic gas concentration and wind observations are generated using a calibrated model derived from real-world drone measurements, providing a more representative testbed that captures atmospheric boundary layer variability. We compare three path planning strategies—preplanned, myopic (short-sighted), and non-myopic (long-term)—and find that non-myopic policies trained via deep reinforcement learning consistently yield more precise and accurate estimates of both source location and emission rate. We further investigate centralized multi-agent collaboration and observe comparable performance to independent agents in the tested single-source scenario. Our results suggest that effective source term estimation depends on correctly identifying the plume and obtaining low-noise concentration measurements within it. Precise localization further requires sampling in close proximity to the source, including slightly upwind. In more complex environments with multiple emission sources, multi-agent systems may offer advantages by enabling individual drones to specialize in tracking distinct plumes. These findings support the development of intelligent, data-driven sampling strategies for drone-based environmental monitoring, with potential applications in climate monitoring, emission inventories, and regulatory compliance.

Impact Statement

Methane is a potent greenhouse gas, making accurate monitoring of its sources essential for effective climate action. This study presents a framework that integrates drone-based measurements with artificial intelligence to improve the detection and quantification of methane emissions. By combining Bayesian inference with atmospheric dispersion modeling and drone-based atmospheric data, the system estimates source locations and emission rates. Reinforcement learning enables the drone to plan adaptive flight paths that maximize information gain in unfamiliar environments. This framework supports the development of intelligent, efficient tools for environmental monitoring, with applications in climate science, emissions reporting, and regulatory compliance.

1. Introduction

Methane (CH_4) is an important greenhouse gas emitted by a range of anthropogenic activities, including fossil fuel extraction, agriculture, and waste management, as well as natural sources such as inland waters,

This research article was awarded Open Data and Open Materials badges for transparent practices. See the Data Availability Statement for details.

© The Author(s), 2026. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (http://creativecommons.org/licenses/by/4.0), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

wild animals, and thawing permafrost (IPCC, 2023). Accurate quantification of CH_4 emissions are essential for climate change mitigation; yet, current national and global inventories remain highly uncertain (Saunois et al., 2020). As identified by Saunois et al. (2024), increasing the density of CH_4 observations at local to regional scales are among the top five priorities for reducing uncertainty in the global CH_4 budget.

To address this challenge, we investigate the use of drones to collect local CH_4 measurements, enabling the estimation of point-source emissions. Several airborne techniques have been developed for quantifying CH_4 emission rates. The tracer flux ratio method involves co-releasing a tracer gas with a known flow rate and comparing its downwind dispersion to that of CH_4. The mass balance method integrates CH_4 concentrations and wind data across a defined control volume to calculate the net flux. Both methods, which are compared in Lunt et al. (2025) as well as Scheutz et al. (2025), require prior knowledge of the source location. Inverse modeling techniques can infer multiple uncertain source parameters by fitting an atmospheric dispersion model to measured atmospheric data. We use a Bayesian inverse modeling approach, which offers a probabilistic framework for jointly estimating both the emission rate and the source location, while explicitly accounting for measurement uncertainty and model error (e.g., Hutchinson et al., 2019b; Park et al., 2021; van Hove et al., 2025).

Drone-based data collection offers spatial flexibility but is resource-intensive. Acquiring informative measurements is particularly difficult in environments with uncertain source locations and dynamic atmospheric conditions (Francis et al., 2022). These constraints underscore the need for intelligent sampling strategies that maximize the informational value of each flight. In this study, we examine the optimization of drone flight paths in unfamiliar environments to improve the estimation of both emission rates and source locations from single-point CH_4 sources.

We compare three path-planning strategies for drone-based CH_4 monitoring: (i) a preplanned lawnmower pattern, (ii) an adaptive short-sighted (myopic) policy that selects actions based on immediate information gain, and (iii) an adaptive long-term (non-myopic) policy trained using deep reinforcement learning. While similar strategies have been explored in previous work (e.g., Loisy and Eloy, 2022; Park et al., 2022; Loisy and Heinonen, 2023; van Hove et al., 2023; van Hove et al., 2024), prior studies often rely on synthetic data generated from simplified models, such as discretized observation spaces or additive white Gaussian noise assumptions, that may not adequately capture the complexity and variability of real-world atmospheric conditions.

In contrast, we generate synthetic observations using an observation model calibrated to replicate CH_4 dispersion from a point source under Arctic conditions. This calibrated model is referred to as the nature run, following the concept used in observing system simulation experiments (OSSEs) (Masutani et al., 2010). In OSSEs, a nature run represents a high-resolution atmospheric simulation that acts as a surrogate for the "true" atmospheric state, enabling the generation of realistic synthetic observations for testing hypothetical observing systems. Our approach adopts this principle but grounds it in field data. Specifically, we tune the observation model using measurements from a drone campaign in Svalbard. Physical parameters are derived directly from field observations, while observation error characteristics are estimated through statistical calibration methods, including maximum likelihood estimation (MLE) and Bayesian model comparison. The resulting nature run provides a controlled, data-driven environment from which synthetic observations are generated to evaluate the different path-planning strategies under varied conditions, such as source location, emission strength, and wind conditions.

To our knowledge, this is the first study to use a data-driven nature run to train and assess adaptive drone sampling strategies for joint estimation of source location and emission rate. Deep reinforcement learning has rarely been applied to this problem (e.g., Park et al., 2022; Lee et al., 2025), and its use with information gain as a reward signal remains especially uncommon—making our approach both novel and technically challenging.

We further explore the role of drone collaboration by comparing single-agent systems (i.e., one drone) with multi-agent systems (i.e., multiple drones). Specifically, we assess whether centralized, coordinated exploration improves the accuracy of source term estimates compared to independent operation. Our objectives are threefold: (i) to generate more realistic synthetic data informed by real-world observations,

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

(ii) to evaluate whether adaptive sampling improves estimation accuracy over preplanned baseline strategies, and (iii) to explore whether collaboration between two drones yields additional benefits over independent operation. Ultimately, our goal is to support the development of more reliable emission inventories and inform targeted mitigation strategies at local to regional scales by improving the accuracy and efficiency of CH_4 source term estimation methods.

2. Methods

We present a framework for estimating the location and emission rate of a single CH_4 source using drone-based observations. Our approach consists of two components: (i) source term estimation (STE) (Hutchinson et al., 2017), which applies Bayesian inverse modeling to infer the source parameters from CH_4 concentration and wind data; and (ii) adaptive informative path planning (AIPP) (Popović et al., 2024), which optimizes drone trajectories to collect the most informative observations in a limited flight time. A schematic overview of the framework and its methodologies is provided in Figure 1.

2.1. Source term estimation

Estimating the location and emission rate of a CH_4 source from drone observations is formulated as an inverse modeling problem. The source parameters of interest cannot be observed directly, but must be inferred from indirect observations of CH_4 concentration and wind.

This relationship is expressed by the observation model:

$$\boldsymbol{d} = \mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\zeta}) + \boldsymbol{\epsilon}, \tag{2.1}$$

where $\boldsymbol{d}$ denotes the observed data, $\boldsymbol{\theta}$ represents the uncertain source parameters to be estimated, and $\boldsymbol{\zeta}$ includes parameters assumed to be known. The forward model $\mathcal{F}(\cdot)$ maps these inputs to predicted observations, while the residual term $\boldsymbol{\epsilon}$ accounts for measurement noise, including effects of flow disturbances caused by the drone (e.g., downwash), errors in the assumed known parameters, limitations of the forward model, and representation errors (van Leeuwen, 2015).

This inverse problem is inherently nonlinear and ill-posed, and is further complicated by sparse, noisy, and intermittent data due to atmospheric turbulence and limited sampling. We adopt a probabilistic framework based on Bayesian inference (Jaynes, 2003) to invert the observation model, Eq. (2.1), and obtain an uncertainty-aware estimate of the uncertain source parameters $\boldsymbol{\theta}$ (Sanz-Alonso et al., 2023).

2.1.1. Observation model

The observational data vector $\boldsymbol{d}_t = [c(\boldsymbol{x},t), v(\boldsymbol{x},t), \phi(\boldsymbol{x},t)]$ consists of instantaneous measurements of CH_4 concentration $c(\boldsymbol{x},t)$ [ppm], horizontal wind speed $v(\boldsymbol{x},t)$ [$\mathrm{m\,s^{-1}}$], and wind direction $\phi(\boldsymbol{x},t)$ [°], recorded at location $\boldsymbol{x} = [x,y,z]$ [m] and time step t [−]. Our goal is to infer the mean emission rate Q [$\mathrm{kg\,h^{-1}}$] and the horizontal source location $\boldsymbol{x}_s = [x_s, y_s]$ [m]. Additionally, we infer two nuisance parameters (Jaynes, 2003): the mean horizontal wind speed V [$\mathrm{m\,s^{-1}}$] and the mean wind direction Φ [°], which are not of primary interest but improve the robustness of the inference. The full vector of uncertain parameters that we seek to infer is thus $\boldsymbol{\theta} = [Q, x_s, y_s, V, \Phi]$.

To facilitate probabilistic inference, we model the residuals between observations and model predictions as Gaussian noise. However, real-world CH_4 concentration and wind data often exhibit non-normal error distributions. To address this mismatch, we apply Box–Cox transformations (Box and Cox, 1964) within the observation model. These transformations, parameterized by λ, map non-normal data from physical space into a transformed observation space where the residuals more closely approximate normality. Separate transformations are applied to CH_4 concentration, wind speed, and wind direction, each with its own optimal λ and transformed observation error standard deviation $\sigma^{(\lambda)}$. To construct the nature run, these parameters are calibrated using real-world drone measurements to replicate realistic measurement noise (see Section 3.1 for their reported values).

We define the transformed observation model as a system of equations evaluated at each time step t:

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

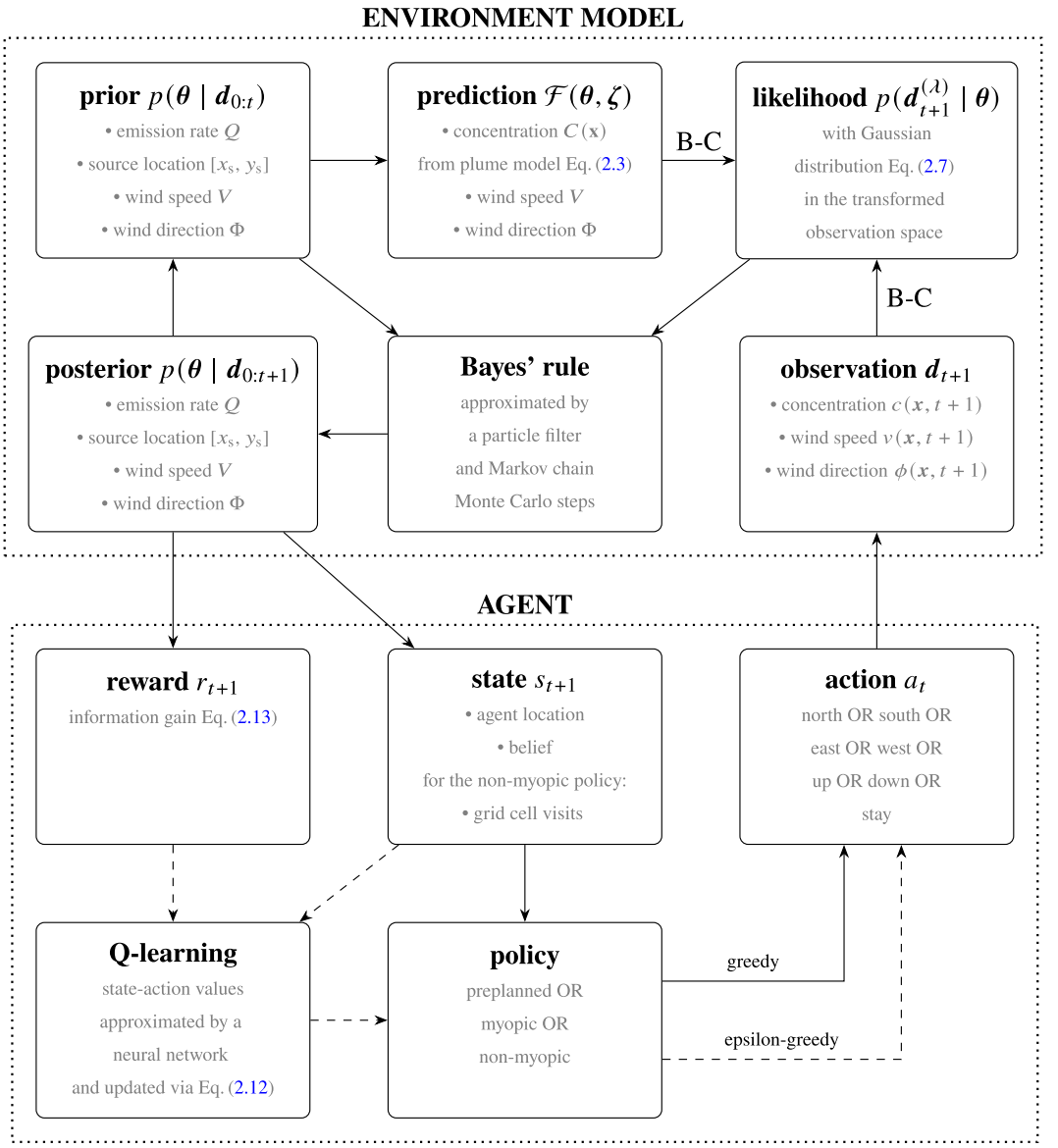


Figure 1. *Schematic overview of the developed framework, consisting of two main components: source term estimation (top box, Section 2.1) and drone path planning strategies (bottom box, Section 2.2). Source estimation uses Bayesian inverse modeling in the Box–Cox transformed observation space, where "B-C" denotes the Box–Cox transformation. Bayesian updating is sequential in the synthetic experiments, with each posterior becoming the prior for the next step. Path planning is guided by the agent (i.e., drone) policy: preplanned (deterministic), myopic (greedy one-step lookahead), or non-myopic (greedy via a trained state-value function). Dashed lines indicate components used only during reinforcement learning training, where a neural network approximates state-action values using epsilon-greedy exploration.*

$$\boldsymbol{d}_t^{(\lambda)} = \begin{cases} c(\boldsymbol{x},t)^{(\lambda)} = C(\boldsymbol{x})^{(\lambda)} + \boldsymbol{\epsilon}_C^{(\lambda)}, \\ v(\boldsymbol{x},t)^{(\lambda)} = V^{(\lambda)} + \boldsymbol{\epsilon}_V^{(\lambda)}, \\ \phi(\boldsymbol{x},t)^{(\lambda)} = \Phi^{(\lambda)} + \boldsymbol{\epsilon}_\Phi^{(\lambda)}, \end{cases} \tag{2.2}$$

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

where $C(\boldsymbol{x})$ [ppm] denotes the temporally averaged CH_4 concentration at the location $\boldsymbol{x}$, and $\boldsymbol{\epsilon}^{(\lambda)} = \left[\epsilon_C^{(\lambda)}, \epsilon_V^{(\lambda)}, \epsilon_\Phi^{(\lambda)}\right]$ represents additive observational and model errors in the transformed space. We assume that these transformed residuals follow a multivariate normal distribution: $\boldsymbol{\epsilon}^{(\lambda)} \sim \mathcal{N}(0, \mathbf{R}_\lambda)$, where $\mathbf{R}_\lambda$ is a diagonal covariance matrix encoding the standard deviations of the transformed observation errors.

The mean concentration field $C(\boldsymbol{x})$ is modeled using an advection–dispersion formulation from Vergassola et al. (2007) that simulates particle transport under turbulent atmospheric conditions:

$$C(\boldsymbol{x}) = \frac{\alpha Q}{4\pi K|\boldsymbol{x}-\boldsymbol{x}_s|} \exp\left(\frac{-(x-x_s)V\sin(\Phi)}{2K}\right) \exp\left(\frac{-(y-y_s)V\cos(\Phi)}{2K}\right) \exp\left(\frac{-|\boldsymbol{x}-\boldsymbol{x}_s|}{\mathcal{L}}\right) + C_0, \tag{2.3}$$

where C_0 [ppm] is the background CH_4 concentration, and K [$m^2 s^{-1}$] represents the mean effective diffusivity, which comprises both turbulent diffusivity and the generally much smaller molecular diffusivity (Stull, 1989). The conversion factor α incorporates the ideal gas law using air temperature T and atmospheric pressure P to facilitate unit conversion between mass concentration [$kg\,m^{-3}$] and mixing ratio [ppm], as well as between temporal units. Additionally, α includes a factor of 2 to account for CH_4 mass conservation, assuming a ground-level source with dispersion over flat terrain. The effective dispersion length scale $\mathcal{L}$ [m] is defined as:

$$\mathcal{L} = \sqrt{\frac{K\tau}{1+\frac{V^2\tau}{4K}}}, \tag{2.4}$$

where τ [s] is the atmospheric lifetime of CH_4.

The vector of assumed known input parameters to the forward model is $\boldsymbol{\zeta} = [\boldsymbol{x}, z_s, \tau, K, C_0, T, P]$, where $z_s = 0$ m is the source height and $\tau = 9.1$ years (Prather et al., 2012) is the atmospheric lifetime of CH_4. The remaining fixed parameters used to construct the nature run are derived from the real-world drone experimental data (see Section 3.1).

2.1.2. Bayesian inference

We adopt a sequential Bayesian inference (also known as filtering) approach (Murphy, 2023; Särkkä and Svensson, 2023) to estimate the uncertain parameters $\boldsymbol{\theta}$ from indirect observations $\boldsymbol{d}$, integrating prior knowledge with new observational data and quantifying uncertainty in the estimates.

In our setting, inference is performed sequentially: each new set of observations—comprising CH_4 concentration, wind speed, and wind direction—triggers a single Bayesian update, allowing the posterior distribution to evolve over time as more data are collected. At each new time step $t+1$, the prior distribution over the unknown parameters $p(\boldsymbol{\theta}|\boldsymbol{d}_{0:t})$ is updated to a posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{d}_{0:t+1})$ conditioned on the new observation(s) $\boldsymbol{d}_{t+1}$ via Bayes' rule:

$$p(\boldsymbol{\theta}|\boldsymbol{d}_{0:t+1}) = \frac{p(\boldsymbol{d}_{t+1}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{d}_{0:t})}{p(\boldsymbol{d}_{t+1}|\boldsymbol{d}_{0:t})}. \tag{2.5}$$

In our plume model, the uncertain parameters $\boldsymbol{\theta}$ are treated as constants over time. Therefore, we apply particle filtering under the assumption of a static model (Chopin, 2002; van Hove et al., 2025). The denominator is the model evidence (also known as marginal likelihood) (MacKay, 2003), acting as a normalizing constant:

$$p(\boldsymbol{d}_{t+1}|\boldsymbol{d}_{0:t}) = \int p(\boldsymbol{d}_{t+1}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{d}_{0:t})\,\mathrm{d}\boldsymbol{\theta}. \tag{2.6}$$

In this sequential framework, the posterior at the time step t serves as the prior at the time step $t+1$. For notational convenience, we define $\boldsymbol{d}_0$ as an empty set, so that $p(\boldsymbol{\theta}|\boldsymbol{d}_0)$ denotes the initial prior. We assume uniform priors over all elements of $\boldsymbol{\theta}$. The prior bounds, specified in Table 1, are deliberately chosen to

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

Table 1. *Uniform prior bounds, training/test scenario bounds, and true values used in the illustrative run for the five uncertain parameters. These parameters define the search space for Bayesian inference and reinforcement learning-based path planning strategies*

Parameter	Prior bounds	Training/test bounds	Illustrative run
$Q\,[\mathrm{kgh^{-1}}]$	[0, 13]	[2.6, 10.3]	3.9
$x_s\,[\mathrm{m}]$	[0, 300]	[60, 240]	150
$y_s\,[\mathrm{m}]$	[0, 300]	[30, 210]	66
$V\,[\mathrm{ms^{-1}}]$	[0, 6]	[1, 5]	2.7
$\Phi\,[^\circ]$	[0, 360]	[140, 220]	200

encompass the bounds of the training and testing domains to avoid biasing the inference process. Additionally, we assume some prior knowledge of the prevailing wind direction, which informs the design of the measurement domain. Specifically, we position the measurement domain slightly downwind of the region where the source is expected to be located, increasing the likelihood of plume detection.

The likelihood term $p(\boldsymbol{d}_{t+1}|\boldsymbol{\theta})$ quantifies the agreement between the observed data and model predictions. In our implementation, the likelihood is modeled as a multivariate Gaussian in Box–Cox transformed observation space (see Appendix B and Eq. [B.3]) following Smith et al., 2010:

$$p\left(\boldsymbol{d}_{t+1}^{(\lambda)}|\boldsymbol{\theta}\right) = \mathcal{N}\left(\boldsymbol{d}_{t+1}^{(\lambda)}|\mathcal{F}(\boldsymbol{\theta},\boldsymbol{\zeta})^{(\lambda)}, \mathbf{R}_\lambda\right), \tag{2.7}$$

where $\mathcal{F}(\boldsymbol{\theta},\boldsymbol{\zeta})^{(\lambda)}$ is the Box–Cox-transformed forward model prediction, and the observation error covariance $\mathbf{R}_\lambda$ is a diagonal matrix containing the square of the error standard deviations $\sigma^{(\lambda)}$ estimated via calibration against field data (see Section 2.4). The transformed likelihood (Eq. [2.7] and Eq. [B.3]) and evidence can be directly substituted into Eq. (2.5), as the Jacobian of the transformation term cancels out owing to its independence from $\boldsymbol{\theta}$ (Jaynes, 2003).

2.1.3. Particle filter

In most real-world scenarios, exact Bayesian inference is computationally intractable due to the complexity and high dimensionality of the posterior distributions. We adopt a Sequential Monte Carlo approach (Chopin and Papaspiliopoulos, 2020; Särkkä and Svensson, 2023) to approximate the posterior distribution of the uncertain parameters $\boldsymbol{\theta}$.

The prior is represented by a weighted ensemble of particles:

$$\left\{\left(\boldsymbol{\theta}^{(j)}, w_t^{(j)}\right)\right\}_{j=1}^{N_e}, \tag{2.8}$$

where $\boldsymbol{\theta}$ denotes the state of a particle j, w_t represents its corresponding weight at the time step t (reflecting its probability given the data up to time step t), and N_e is the total number of particles in the ensemble. The weights are self-normalized such that $\sum_{j=1}^{N_e} w_t^{(j)} = 1$ (Murphy, 2023).

At each iteration, the particle weights are updated using importance sampling based on the transformed likelihood, Eq. (2.7), of the new observation:

$$\bar{w}_{t+1}^{(j)} \propto p\left(\boldsymbol{d}_{t+1}^{(\lambda)}|\boldsymbol{\theta}^{(j)}\right) w_t^{(j)}. \tag{2.9}$$

The weights $\bar{w}_{t+1}^{(j)}$ are subsequently self-normalized to approximate the posterior $p(\boldsymbol{\theta}|\boldsymbol{d}_{0:t+1})$.

A common issue in particle filtering is degeneracy, where a small number of particle paths dominate the weight distribution (Murphy, 2023), leading to poor representation of the posterior distribution. To mitigate this, we deploy the resample-move algorithm (Gilks and Berzuini, 2001; Doucet and Johansen,

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

2009), which combines resampling with a Markov Chain Monte Carlo (MCMC) move step based on the Metropolis–Hastings algorithm.

To balance exploration and exploitation, we employ an adaptive step size for the MCMC proposals. The step size is set to the bandwidth of the dynamic prior particle distribution from time t, computed using Silverman's rule, Eq. (D.2). This adaptive proposal distribution helps prevent premature convergence and supports robust posterior estimation. Reflective boundary conditions are applied to ensure that proposed particles remain within the prior bounds.

Online path planning requires data processing during collection, necessitating fast updates. As such, we use $N_e = 1{,}000$ particles and a single MCMC move step per iteration, updating the posterior after each new observation $\boldsymbol{d}_{t+1}$.

2.2. *Adaptive informative path planning*

Optimizing the drone's flight path for STE is formulated as a sequential decision-making problem under uncertainty. At each time step, the drone must decide where to move next in order to collect the most informative observations. We model this task as a Partially Observable Markov Decision Process (Kochenderfer et al., 2022), where the agent, that is, the drone, interacts with the environment through a cycle of actions, observations, and belief updates.

In this framework, the agent does not have direct access to the true state of the environment $\boldsymbol{\theta}$, but instead maintains a belief over the uncertain parameters, updated via the particle filter described in Section 2.1.3. The belief state at time step t, denoted s_t, includes a representation of the distribution over the source parameters $p(\boldsymbol{\theta}|\boldsymbol{d}_{0:t})$, as well as assumed known parameters such as the drone's current position $\boldsymbol{x}_t$.

At each time step, the agent selects an action a_t from a discrete set of movement options in a three-dimensional domain: one step in any cardinal direction (north, south, east, and west), vertical movement (up and down), or remaining stationary. After executing the action, the agent receives new observations $\boldsymbol{d}_{t+1}$, a numerical reward r_{t+1}, and transitions to a new belief state s_{t+1}. The episode terminates after a fixed number of steps, simulating battery life.

The agent's objective is to maximize the expected cumulative reward, defined as the sum of discounted future rewards over a finite horizon:

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \ldots = \sum_{k=t+1}^{t_{\text{final}}} \gamma^{k-t-1} r_k, \tag{2.10}$$

where $\gamma \in [0, 1]$ is the discount factor controlling the agent's planning horizon, and t_{final} is the final time step of the episode. In our setting, the reward function is based on information gain—specifically, the reduction in entropy of the posterior relative to the dynamic prior (see Section 2.3). This encourages the agent to collect observations that most effectively reduce uncertainty.

To explore different planning horizons, we evaluate both myopic (short-sighted) and non-myopic (long-term) policies, as described in the following section.

2.2.1. *Myopic and non-myopic policies*

The agent's decision-making is governed by a policy π, which maps belief states to actions. We consider two classes of policies that differ in their planning horizon: myopic and non-myopic. In a myopic policy, the agent selects actions based solely on the expected immediate reward. This corresponds to setting the discount factor $\gamma = 0$ in Eq. (2.10), reducing the return to $G_t = r_{t+1}$. In contrast, a non-myopic policy considers the long-term consequences of actions by using a discount factor γ close to 1, thereby incorporating future rewards into the decision-making process.

To guide action selection, we estimate the action-value function $q_\pi(s, a) = \mathbb{E}_\pi[G_t | s_t = s, a_t = a]$, which represents the expected return when taking action a in belief state s and following the policy π thereafter. A greedy agent selects the action that maximizes the function: $\pi(s) = \arg\max_a q(s, a)$.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

The optimal action-value function $q_*(s,a)$ satisfies the Bellman optimality equation:

$$q_*(s,a) = \mathbb{E}\left[r_{t+1} + \gamma \max_{a'} q_*(s_{t+1}, a') | s_t = s, a_t = a\right]. \tag{2.11}$$

In practice, computing this expectation exactly is infeasible due to the high dimensionality of the belief space and the stochastic nature of observations. We therefore rely on sample-based approximations.

For the myopic policy with $\gamma = 0$, the Bellman optimality equation simplifies to: $q_*(s,a) = \mathbb{E}[r_{t+1}|s_t = s, a_t = a]$, which we estimate using Monte Carlo sampling. Specifically, we draw samples from the current belief state, simulate the outcome of each candidate action, and compute the expected immediate reward. To maintain computational efficiency during online decision-making, we limit the number of samples to one per candidate action, trading off accuracy for speed.

In contrast, the non-myopic policy requires estimating the long-term value of actions. We approximate the action-value function using a parameterized function $\hat{q}(s,a;\boldsymbol{\omega})$, implemented as a deep neural network with weights $\boldsymbol{\omega}$. The training of this network through deep reinforcement learning is described in the following section.

2.2.2. Reinforcement learning

To implement the non-myopic policy, we use reinforcement learning (RL) (Sutton and Barto, 2018), specifically deep Q-learning (Mnih et al., 2015), for the approximation of the optimal action-value function $q_*(s,a)$. This is achieved using a feedforward neural network parameterized by weights $\boldsymbol{\omega}$, denoted by $\hat{q}(s,a;\boldsymbol{\omega})$. The network is implemented in Python using TensorFlow (Martín et al., 2015).

The architecture consists of three fully connected hidden layers with 512, 256, and 128 units, respectively, each using rectified linear unit activation functions. The output layer is linear, with one node per action, that is, seven nodes for navigation in a three-dimensional (3D) domain, representing the estimated action values $\hat{q}(s,a;\boldsymbol{\omega})$.

The input to the network includes a discretized belief map over the source location and wind direction, the drone's current position, and a representation of previously visited observation locations within the domain (see Appendix E).

Training is conducted over 25,000 iterations, with 12 mini-batch updates per iteration. Each update uses 32 transitions sampled uniformly from a replay buffer of 10,000 transitions.

The Q-values are updated using the standard Q-learning rule:

$$\hat{q}(s_t, a_t; \boldsymbol{\omega}) \leftarrow \hat{q}(s_t, a_t; \boldsymbol{\omega}) + \eta\left[r_{t+1} + \gamma \max_a \hat{q}_{\text{target}}(s_{t+1}, a; \boldsymbol{\omega}^-) - \hat{q}(s_t, a_t; \boldsymbol{\omega})\right], \tag{2.12}$$

where $\eta = 0.0001$ is the learning rate, optimized using the Adam optimizer (Diederik and Jimmy, 2015). The target Q-value is computed using a separate target network $\hat{q}_{\text{target}}$, which is synchronized with the main network every 500 iterations to improve training stability.

Exploration is managed using an ϵ-greedy policy. The exploration rate ϵ starts at 1.0 and decays toward 0.01 over the course of training, encouraging exploration early on and exploitation of learned strategies later. While training was conducted for 25,000 episodes, we observed that cumulative reward sometimes deteriorated during the final stages of training. To ensure robust evaluation, we then selected the policy from an earlier, better-performing iteration.

2.2.3. Centralized multi-agent system

In a centralized multi-agent framework, multiple agents collaboratively maintain a joint belief over the environment. Since agents' actions are interdependent, optimal decision-making requires selecting joint actions from the combined action space, which grows exponentially with the number of agents. To manage this complexity, we restrict our study to two agents operating in a two-dimensional domain. This simplification reduces the size of the joint action space and makes centralized coordination computationally feasible for our synthetic experiments.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

In the collaborative setting, the joint belief is updated sequentially at each time step using the observational data from each agent, resulting in two inference steps, one for each agent. For comparison, we also evaluate a non-collaborative scenario in which each drone follows its own trajectory and maintains an independent belief.

In addition, we explore an alternative inference approach where the joint belief is updated in a single inference step using aggregated observational data. Specifically, at each time step, the observations from both agents are concatenated into a single vector and assimilated jointly, rather than sequentially.

Training of the centralized multi-agent RL (Albrecht et al., 2024) policy with joint assimilation is conducted over three training sessions, each comprising 25,000 iterations. In the first session, the exploration rate ϵ decays from 1.0 to 0.01. In the second and third sessions, ϵ decays from 0.5 to 0.01. Subsequently, the policy with sequential assimilation is initialized from the joint assimilation policy and retrained over an additional 25,000 iterations, with ϵ decaying from 0.5 to 0.01.

2.3. Information entropy

Our primary objective is to obtain accurate and precise estimates of the source term parameters. To this end, we define a reward function based on information entropy $\mathcal{H}(\cdot)$, which quantifies the uncertainty associated with these parameters. This approach incentivizes actions that reduce uncertainty in the agent's belief state.

Entropy reflects how uncertain we are about a random variable; the higher the entropy, the less we know about its value (MacKay, 2003). We prioritize entropy over variance, as variance can be misleading, particularly in multimodal distributions where high variance may still correspond to informative beliefs. Entropy is maximal for uniform distributions, such as our priors, and decreases as the distribution becomes more constrained. We define the reward at each time step as the reduction in entropy between successive belief states, that is, information gain (Lindley, 1956):

$$r_{t+1} = \mathcal{H}(s_t) - \mathcal{H}(s_{t+1}). \tag{2.13}$$

For continuous distributions, entropy generalizes to differential entropy, defined as:

$$\mathcal{H}(s) = -\int p(\boldsymbol{\theta}) \log p(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta}, \tag{2.14}$$

where $p(\boldsymbol{\theta})$ denotes the probability density over the unknown parameters $\boldsymbol{\theta}$. We use the natural logarithm base e, yielding entropy in units of [nats].

In our framework, the posterior distribution $p(\boldsymbol{\theta})$ is approximated by a weighted particle set, Eq. (2.8). The integral in Eq. (2.14) is then approximated via Monte Carlo integration:

$$\mathcal{H}(s) \approx -\sum_{j=1}^{N_e} w^{(j)} \log p\left(\boldsymbol{\theta}^{(j)}\right), \tag{2.15}$$

where $\boldsymbol{\theta}^{(j)}$ are the particles and $w^{(j)}$ their associated weights.

Following Fischer and Tas (2020), we employ kernel density estimation (KDE) (Gisbert, 2003) to calculate differential entropy from a weighted particle set. KDE provides a smooth approximation of the posterior density from the weighted particle ensemble. Further implementation details are provided in Appendix D.

Several previous STE studies (e.g., Hutchinson et al., 2019a; Hutchinson et al., 2020; Park et al., 2021) approximate entropy using only the particle weights:

$$\mathcal{H}(s) \approx -\sum_{j=1}^{N_e} w^{(j)} \log w^{(j)}. \tag{2.16}$$

This formulation neglects the spatial distribution and density of particles in the parameter space. As noted by Boers et al. (2010), such an approximation is sensitive to resampling: since resampling assigns uniform

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

weights ($w^{(j)} = 1/N_e$), it can artificially reduce the entropy to a constant value, regardless of the actual distribution of particles. In contrast, the KDE-based approach incorporates both particle weights and their positions, yielding a more accurate and informative measure of uncertainty. While this method is computationally more demanding, it avoids misleading entropy estimates.

2.4. Nature run

In this study, we evaluate different path-planning strategies through synthetic experiments, where observations are generated by a nature run that serves as a proxy for "true" plume dispersion in an Arctic environment. The nature run is a calibrated observation model (see Section 2.1.1) constructed using real-world data from a field experiment. This experiment involved drone-based measurements at a legacy borehole in Adventdalen, Svalbard, which is normally sealed and inactive. The borehole—well 1967–1971—was originally drilled during coal exploration by Store Norske Spitsbergen Kulkompani, and reaches a depth of 106 m. The well intersects a shallow gas accumulation at the base of the permafrost within the Lower Cretaceous Helvetiafjellet Formation. Between 1967 and 1975, the well intermittently produced over 2.5 million cubic meters of CH_4. The borehole is situated in a valley where the permafrost is ice-saturated and acts as an effective cryogenic seal, trapping gas beneath it (Birchall et al., 2023).

Following a flood and subsequent freezing period, structural damage to the borehole was suspected to have initiated a leak. We drilled five shallow holes in the surface ice to release trapped CH_4 and allowed ~30 min for flow stabilization prior to measurement.

The field experiment was performed on 1 February 2025, using a DJI M300 RTK quadcopter (DJI, China) equipped with an AERIES MIRA Strato LDS methane gas sensor (Aeries Technologies, USA) and a Trisonica Sphere 3D sonic anemometer (LI-COR, USA) (see Figure 2). Atmospheric observations were collected over a 150 m × 200 m area surrounding the source. The drone followed a preplanned lawnmower flight pattern at altitudes of ~2, ~4, and ~6 m above ground level, moving at a speed of 1.4 m s^{-1}.

The gas sensor measured CH_4 concentrations via mid-infrared laser spectroscopy at 1 Hz, while the sonic anemometer recorded wind speed and direction at 20 Hz. Wind measurements were corrected for drone motion (pitch, roll, yaw, and translation).

The collected data were used to calibrate the following parameters in the nature run: (i) the effective diffusivity K using the Businger–Dyer relationship (Appendix A), (ii) the background concentration C_0,



Figure 2. *(a) The drone used in the field experiment, equipped with a methane sensor mounted underneath, and a 3D sonic anemometer on top. The pole with reflectors marks the location of the leaking borehole. Methane was released by drilling five shallow holes through the ice to access the underlying water. Credit: Gina Schulz. (b) The drone is collecting atmospheric observations near the borehole.*

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

atmospheric temperature T, and pressure P, (iii) the Box–Cox transformation parameters λ and the transformed observation error standard deviations $\epsilon^{(\lambda)}$ for concentration, wind speed, and wind direction in Eq. (2.2). In addition, an estimate of the emission rate from the leak is provided.

For wind data, we calibrate the transformation parameters by applying MLE to wind speed and direction measurements from the field experiment (see Appendix B). Following the observation model, Eq. (2.2), we directly infer the mean wind speed V and direction Φ from the observed quantities $v(\boldsymbol{x},t)$ and $\phi(\boldsymbol{x},t)$, allowing us to apply MLE directly to the observed data to estimate the Box–Cox parameters. In contrast, for concentration $c(\boldsymbol{x},t)$, the inference involves hidden parameters in $\boldsymbol{\theta}$ that are nonlinearly and indirectly related to the temporally averaged concentration $C(\boldsymbol{x})$. Consequently, we adopt a more sophisticated particle-based Bayesian model comparison approach to calibrate the transformation parameters for the concentration error model (see Appendix C).

2.5. Synthetic experiments

The synthetic experiments are designed to reflect the conditions of the field experiment. The simulation domain is $330\,\text{m} \times 330\,\text{m}$, discretized into an 11×11 grid. The resolution of 30 m was chosen as a compromise: a finer grid would offer greater spatial precision for drone navigation but increase the computational costs of RL policy training, while a coarser grid would reduce computational costs at the expense of navigation flexibility and possibly source localization accuracy. For single-agent test cases, observations are collected at altitudes of 2 m, 4 m, and 6 m above ground level. We assume some prior knowledge of the prevailing wind direction, as is typically available in field deployments, and initialize the drone downwind of the source at position $[0,300,6]\,\text{m}$. We evaluate flight durations of 108, 54, and 27 steps. To reduce computational costs in multi-agent test cases, data collection is restricted to a fixed altitude of 2 m, with both drones starting at $[0,300,2]\,\text{m}$. We consider flights of 27 steps.

The RL policy is trained offline using domain randomization. In each training episode, the true values for the uncertain parameters $\boldsymbol{\theta}$ to be inferred are uniformly sampled from the ranges specified in Table 1. Policy performance is assessed across the same parameter ranges using a testbed of 25,000 simulated flights. During both training and testing, synthetic observations are generated from the nature run. Specifically, we sample from the transformed observation model defined in Eq. (2.2), and then apply an inverse Box–Cox transformation to recover observations in physical space. This model incorporates: (i) the forward model, which takes as input the sampled true values of $\boldsymbol{\theta}$ and the assumed known input parameters $\boldsymbol{\zeta}$ (as specified in Sections 2.1.1 and 3.1); (ii) Box–Cox transformation parameters λ; and (iii) a Gaussian error model in the transformed observation space, with calibrated noise parameters $\sigma^{(\lambda)}$ detailed in Section 3.1.

2.5.1. Performance metrics

Policy performance is assessed using entropy-based and probabilistic scoring metrics. Continuous Ranked Probability Score (CRPS) (Hersbach, 2000) is a proper scoring rule (Gneiting and Raftery, 2007) that evaluates the quality of probabilistic forecasts by comparing the predicted distribution to the true value. Lower CRPS values indicate better performance, with a score of zero corresponding to a perfectly sharp and accurate prediction. While entropy measures the precision of the posterior distribution, CRPS captures both precision and accuracy but requires access to the true value.

We evaluate policy performance using the following metrics: (i) Cumulative information gain, also referred to as entropy reduction, denoted by $\Delta\mathcal{H}[\text{nats}]$. This metric quantifies the total reduction in uncertainty over the course of the flight from the initial prior at $t = 0$ to the final posterior. (ii) CRPS for emission rate, denoted by $\text{CRPS}_Q\left[\text{kg}\,\text{h}^{-1}\right]$. This metric evaluates the accuracy and precision of the posterior distribution for the emission rate Q. (iii) CRPS for source location, denoted by $\text{CRPS}_{\boldsymbol{x}_s}[\text{m}]$. This is computed as the CRPS of the Euclidean distance between the estimated and true source location $\boldsymbol{x}_s$. (iv) Normalized CRPS, denoted by $\text{CRPS}_{\text{norm}}[-]$, which assesses the improvement of the posterior over the initial prior, averaged across both Q and $\boldsymbol{x}_s$. It is noteworthy that normalized metrics with a value >1 indicate degraded performance relative to the initial prior.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

3. Results

3.1. Nature run

The field experiment provided observations of CH_4 concentration and wind conditions over an emission source (see Figure 3). Methane concentrations close to the source exhibited sharp peaks, suggesting high temporal variability in the emission rate. We also saw bubbles rising intermittently, further supporting the idea that emissions were not steady. Wind speed and direction also varied considerably throughout the experiment.

From the CH_4 concentration data, we estimated the background concentration as $C_0 = 2.09\,\text{ppm}$. The mean temperature and pressure during the flights were $T = 274\,\text{K}$ and $P = 1020\,\text{hPa}$, respectively. Wind measurements at three altitudes were used to estimate an effective diffusivity of $K = 1\,\text{m}^2\,\text{s}^{-1}$, based on the vertical diffusivity profile derived from the Businger–Dyer relationship (see Appendix A). We note that this is a rough estimate based on anemometer observations on a moving drone.

To calibrate the Box–Cox transformation parameters for the wind data error models, we applied MLE to wind speed and direction measurements from the field experiment (see Appendix B).

For wind speed, the optimal transformation parameter was $\lambda = 0.38$, resulting in a transformed error standard deviation of $\sigma_v^{(\lambda)} = 0.83$. This indicates moderate deviation from normality: the transformation compresses high values, while low values are slightly stretched, yielding a distribution that more closely approximates normality. Although derived from a single field experiment, the transformation parameters are assumed to generalize under similar turbulence conditions and are applied to the residuals in the observation model under all true parameter configurations. Figure F1 in Appendix F illustrates representative synthetic wind speed observations under the selected Box–Cox transformation parameters in physical space.

For wind direction, we assumed a symmetric error model in which deviations from the mean direction are treated equally probable in both clockwise and counterclockwise directions. Under this assumption, the optimal transformation was the identity ($\lambda = 1.00$) corresponding to normality, and the error standard deviation of the measurements was $\sigma_\phi^{(\lambda)} = 98.09\,^\circ$, indicating substantial variability. To handle the circular nature of wind direction, the normal distribution is folded over the $0 - 360\,^\circ$ range using a modulus operation. Figure F1 in Appendix F illustrates representative synthetic wind direction observations under the selected Box–Cox transformation parameters in physical space.

For CH_4 concentration data, we performed Bayesian model comparison (see Appendix C) to identify suitable Box–Cox transformation parameters. As shown in Figure 4a, several combinations of the transformation parameter λ and the transformed error standard deviation $\sigma_c^{(\lambda)}$ yielded similarly high model evidence. Values of λ near zero correspond to transformations that resemble a logarithmic scale, leading to a heteroscedastic error model. This transformation strongly compresses high CH_4 concentration values,

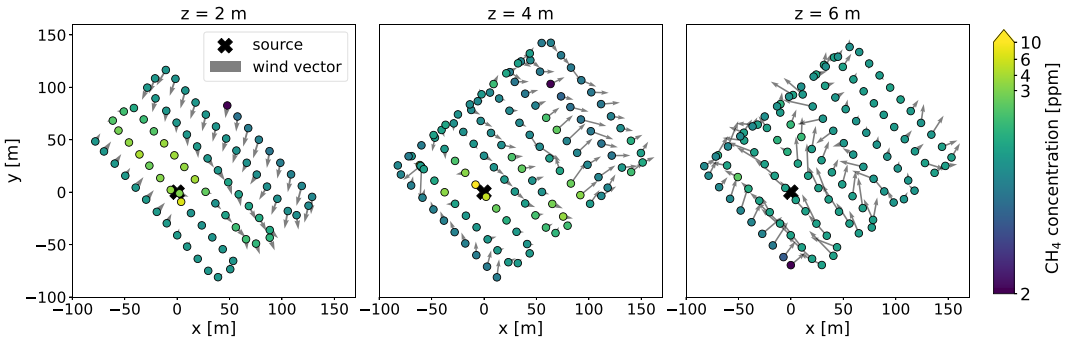


Figure 3. *Methane (CH_4) concentrations observed during the field experiment. The drone followed a preplanned lawnmower flight path at ~2, ~4, and ~6 m above ground level. Each sub-figure corresponds to a single flight leg at constant altitude. Wind vectors are shown as quivers, with lengths scaled to* 20% *of the axis length for visual clarity. All data are resampled to* 0.1 Hz.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

reducing the influence of extreme outliers and helping to normalize long-tailed distributions. While negative values for λ can further flatten the scale for large values, they are extremely heteroscedastic, which may overemphasize low concentrations and complicate interpretability. Although our CH_4 concentration data do not include values near zero, we prioritized robustness. We therefore selected a log transformation ($\lambda = 0$), which offers stable behavior and intuitive scaling. This choice effectively compresses large spikes without distorting the lower tail. Given $\lambda = 0$, we selected the log transformation with a transformed standard deviation of $\sigma_c^{(\lambda)} = 0.15$, which yielded the highest model evidence among the tested log-scale configurations. Figure 4b illustrates representative synthetic CH_4 concentration observations generated under this setting in physical space. The variability in the synthetic observations increases with higher predicted concentrations from the forward model. Although this variability is not explicitly dependent on the drone's proximity to the source, it is implicitly spatially correlated: turbulent observations near the plume core, where predicted mean concentrations $C(\boldsymbol{x})$ are elevated, exhibit greater variability than those farther from the core, where $C(\boldsymbol{x})$ is lower.

Bayesian inference was applied to the field experiment data using the particle filter configuration described in Appendix C. To account for variability introduced by different mini-batch divisions, we aggregated results from 100 independent outer ensemble simulations with the particle filter, each with an inner ensemble of $N_e = 1{,}000$ particles. The final posterior mean ($\pm$ 1 standard deviation) estimate of the emission rate, with the source location fixed at the true position, was $Q = 2.2 \pm 0.6\,\mathrm{kg\,h^{-1}}$. This result suggests that the borehole is leaking, and consequently, repairs will be performed.

3.2. *Single-agent framework*

3.2.1. *Illustrative flights*

To illustrate the behavior of different path planning strategies, we present illustrative flight paths consisting of 108 steps in Figure 5. These were generated using a specific set of true values for the uncertain parameters, as defined in Table 1. The selected examples correspond to runs in which both

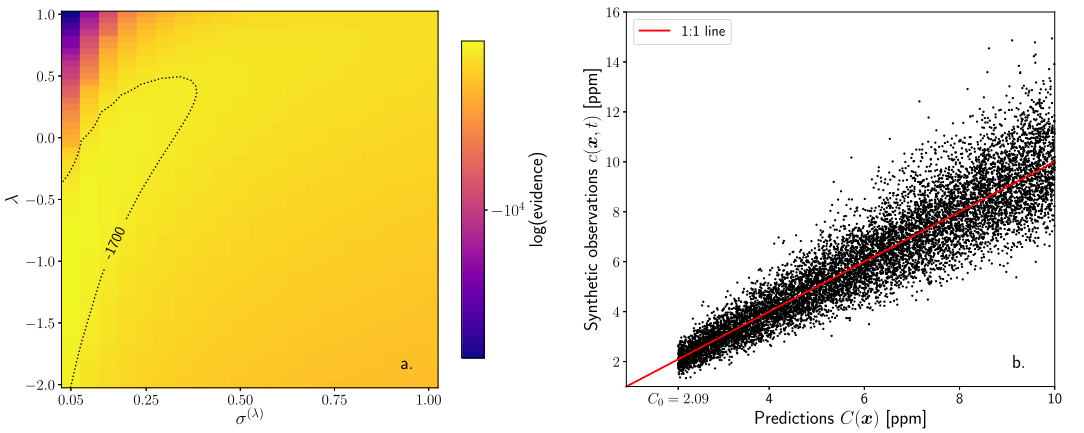


Figure 4. *(a) Logarithm of Bayesian model evidence for different settings of the Box–Cox transformation parameter λ and the corresponding transformed error standard deviation parameter $\sigma_C^{(\lambda)}$ for methane concentration data. Higher model evidence values indicate transformation settings that yield an observation model in the nature run that more closely matches the observed data from the field campaign. (b) Example turbulent synthetic methane concentration observations $c(\boldsymbol{x},t)$ obtained via the inverse Box–Cox transformation of $C(\boldsymbol{x})^{(\lambda)} + \epsilon_C^{(\lambda)}$, with $\lambda = 0$ and $\epsilon_C^{(\lambda)} \sim \mathcal{N}\left(0, \sigma_C^{(\lambda)}\right)$, where $\sigma_C^{(\lambda)} = 0.15$. The predicted mean concentrations $C(\boldsymbol{x})$ are generated by the forward model, Eq. (2.3), assuming a background concentration of $C_0 = 2.09\,\mathrm{ppm}$.*

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

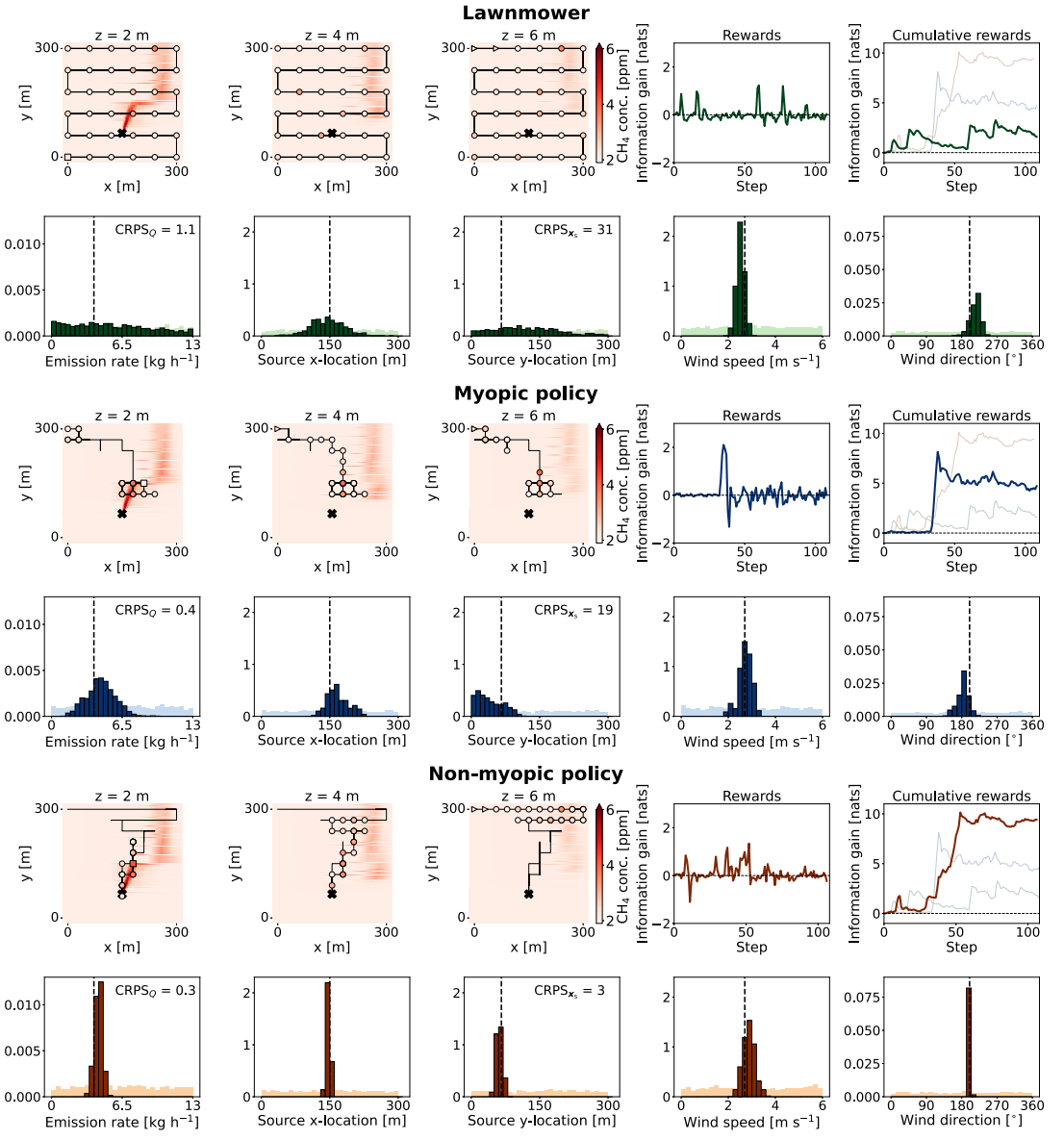


Figure 5. *Illustrative flight paths for the lawnmower flight plan (top), myopic flight plan (middle), and non-myopic flight plan (bottom). The true parameters are* $\boldsymbol{\theta} = [Q = 3.9\,\mathrm{kg\,h^{-1}}, x_s = 150\,\mathrm{m}, y_s = 66\,\mathrm{m}, V = 2.7\,\mathrm{m\,s^{-1}}, \Phi = 200\,^\circ]$. *The expected mean concentration field, following Eq. (2.3), is shown as a background heatmap, with the true source location marked by a cross. The flight paths are overlaid, along with the obtained noisy observations depicted by colored nodes: the first two observations are shown as a triangle, the final as a square, and intermediate observations as circles. The rewards and cumulative rewards for each flight path are plotted. The cumulative rewards of the other flight paths are shown by a thin line for comparison. Histograms display the initial uniform prior in light shading, the final posterior in dark shading, and the true value depicted by a dashed line.*

information gain and CRPS values closely match the median performance across 25,000 simulations conducted with this parameter configuration (see Table 2). These examples are intended to be representative, providing qualitative insight into typical agent behavior.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

Table 2. *Performance metrics for the lawnmower, myopic, and non-myopic flight strategies across 3D and 2D domains, and episode lengths of 108, 54, and 27 steps. Metrics include: (i) cumulative information gain or entropy reduction ($\Delta\mathcal{H} = \mathcal{H}(s_0) - \mathcal{H}(s_{\text{final}})$), (ii) Continuous Ranked Probability Score (*CRPS*) for the emission rate (*CRPS_Q*), (iii)* CRPS *for the Euclidean distance from the true source location (*CRPS_{x_s}*), and (iv)* CRPS *ratio of the posterior compared to the initial prior, averaged over both Q and $\boldsymbol{x}_s$ (*$\text{CRPS}_{\text{norm}}$*). Results are aggregated over 25,000 synthetic test scenarios and reported as median values, followed by the 5th and 95th percentiles in between square brackets. Lower* CRPS *values indicate better performance*

	$\Delta\mathcal{H}$ [nats]	CRPS_Q [kgh^{-1}]	CRPS_{x_s} [m]	$\text{CRPS}_{\text{norm}}$ [−]
Single-agent | 3D domain				
108 steps | illustrative run				
Lawnmower	2.8 [1.1,4.0]	0.8 [0.5,1.7]	41 [29, 79]	0.46 [0.32,0.83]
Myopic	4.5 [0.2,9.9]	0.7 [0.2,2.2]	38 [4,114]	0.44 [0.09,1.14]
Non-myopic	9.8 [7.0,11.2]	0.2 [0.1,2.1]	4 [2,53]	0.10 [0.05,0.84]
108 steps				
Lawnmower	2.2 [0.3,6.0]	1.1 [0.6,3.2]	53 [15,109]	0.73 [0.31,1.50]
Myopic	5.5 [0.2,12.0]	1.0 [0.2,3.4]	32 [5,109]	0.62 [0.14,1.56]
Non-myopic	8.9 [4.6,14.9]	0.6 [0.1,3.6]	10 [2,85]	0.29 [0.08,1.60]
54 steps				
Myopic	2.0 [0.1,9.6]	1.2 [0.3,3.4]	74 [11,122]	0.88 [0.22,1.51]
Non-myopic	8.2 [0.6,14.1]	0.8 [0.2,3.7]	14 [3,104]	0.41 [0.10,1.64]
27 steps				
Myopic	0.3 [0.1,6.8]	1.3 [0.5,3.0]	90 [26,126]	0.97 [0.40,1.42]
Non-myopic	7.5 [0.2,14.1]	1.1 [0.2,4.0]	28 [4,118]	0.64 [0.13,1.75]
Multi-agent | 2D domain | 27 steps				
No collaboration				
Myopic	1.2 [0.2,8.4]	1.2 [0.4,3.4]	77 [13,122]	0.89 [0.26,1.54]
Non-myopic	7.8 [1.0,14.0]	0.8 [0.2,3.7]	19 [4,102]	0.45 [0.11,1.67]
Collaboration				
Myopic	0.9 [0.1,7.7]	1.3 [0.4,3.4]	79 [15,123]	0.90 [0.28,1.54]
Non-myopic	7.4 [0.7,13.6]	0.9 [0.2,3.8]	18 [4,102]	0.46 [0.12,1.66]

Lawnmower. The lawnmower strategy follows a preplanned, systematic sweep of the domain without adaptive decision-making. In the illustrative run, the agent collects observations at 108 evenly spaced locations, resulting in broad posterior distributions for the emission rate and source location. In contrast, wind speed and direction are estimated more precisely, with posteriors concentrated around the true values.

Myopic policy. The myopic policy selects each action based on immediate expected information gain, using a single Monte Carlo sample per candidate action. In the illustrative run, the agent remains within regions of high concentration but avoids sampling directly at or upwind of the source. Compared to the lawnmower strategy, this approach yields a higher cumulative information gain (4.7 nats versus 1.6 nats), and the posterior estimates of emission rate and source location are more constrained, resulting in lower CRPS values.

Non-myopic policy. The non-myopic flight path is governed by an RL policy trained to maximize long-term information gain. In the illustrative run, the agent initially remains downwind at an altitude of 6 m, traveling approximately crosswind relative to the inferred prevailing wind. It then descends and

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

approaches the source, including upwind sampling, which further reduces uncertainty in the source location. Compared to the myopic and lawnmower strategies, the non-myopic policy yields higher cumulative information gain (9.4 nats) and produces more precise posterior estimates of both the emission rate and source location, as reflected in lower CRPS values.

3.2.2. Performance statistics

Table 2 summarizes key performance metrics (defined in Section 2.5.1), including entropy reduction, CRPS scores for emission rate and source location, and normalized CRPS. All results are reported as medians, accompanied by the 5th–95th percentile range to indicate variability across the testbed.

Although agents are trained to maximize information gain, our primary evaluation criterion is posterior predictive performance, quantified using the CRPS. Unlike entropy-based metrics, which quantify belief constraint, CRPS directly reflects the quality of posterior predictions in terms of accuracy and precision (Hersbach, 2000). Figure 6 presents a pairwise comparison matrix showing the proportion of 25,000 testbed simulations in which each strategy achieves a lower $\mathrm{CRPS}_{\mathrm{norm}}$, signifying superior predictive performance compared to its counterparts.

Failure to learn. Across all flight strategies and durations, entropy reduction $\Delta\mathcal{H}$ across testbed simulations deviates from a Gaussian distribution, instead exhibiting a bimodal structure. One broad peak reflects substantial information gain, while a sharp peak near 0–1 nats corresponds to minimal learning. The height of this sharp peak varies by strategy and flight duration. For the lawnmower strategy with 108 steps, the flights fall within this low-information category defined by $\Delta\mathcal{H} < 1$. The myopic policy shows increasing rates of low-information flights with shorter durations: 22%, 44%, and 70% for 108, 54, and 27 steps, respectively. In contrast, non-myopic policies yield substantially lower rates: 2%, 6%, and 15%.

Comparison of flight strategies. Across all flight durations, the non-myopic policy consistently outperforms both lawnmower and myopic strategies. With 108 steps, it achieves the highest information gain (8.9 nats) and the lowest median CRPS values for both emission rate ($0.6\,\mathrm{kg\,h^{-1}}$) and source location (10 m). These results are further supported by the high outperformance fractions shown in Figure 6.

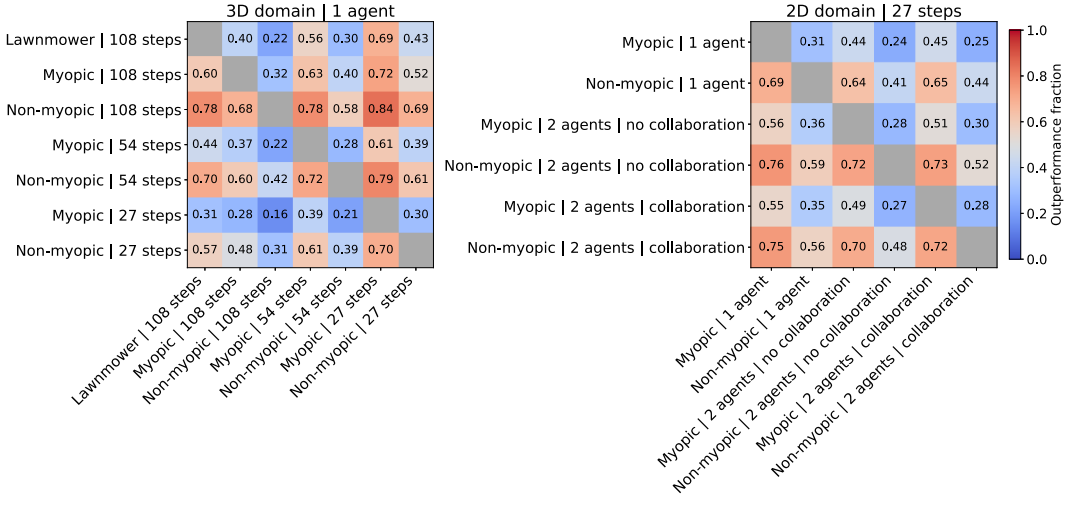


Figure 6. *Annotated matrix illustrating pairwise outperformance fractions between flight strategies and durations. Each cell represents the proportion of simulation runs in which the strategy and duration indicated by the row achieve a lower* $\mathrm{CRPS}_{\mathrm{norm}}$ *than the corresponding configuration in the column. Values are based on 25,000 synthetic testbed runs for each configuration and reflect comparative posterior accuracy and precision across configurations.*

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

The lawnmower strategy shows limited learning, with a low entropy reduction of 2.2 nats over 108 steps and comparatively high CRPS values. The myopic policy performs moderately well, improving over the lawnmower strategy but falling short of the non-myopic performance in all median performance metrics.

Due to computational constraints in online adaptive path planning, the myopic policy was executed using a single Monte Carlo sample per candidate action. To assess the potential for improvement, we also evaluated the myopic strategy with five Monte Carlo samples per candidate action for the 108-step scenario. This increased $\Delta\mathcal{H}$ from 5.5 nats to 5.9 nats and reduced $\mathrm{CRPS}_{\mathrm{norm}}$ from 0.62 to 0.54. While this demonstrates a modest improvement, it remains inferior to the performance of the non-myopic strategy. Although increasing the number of Monte Carlo samples improves policy performance, such an approach is computationally prohibitive in online deployment. In contrast, non-myopic policies require substantial resources for offline training, but evaluating the trained neural network during online execution is computationally efficient.

Impact of flight duration. Shorter flight durations lead to degraded performance, particularly for the myopic policy. At 54 steps, entropy reduction $\Delta\mathcal{H}$ drops from 5.5 nats to 2.0 nats, and at 27 steps, it falls further to 0.3 nats. The corresponding CRPS values also worsen. Despite reduced durations, the non-myopic agents maintain relatively strong performance. At 27 steps, the non-myopic agent achieves $\Delta\mathcal{H} = 7.5$ nats and $\mathrm{CRPS}_{\mathrm{norm}} = 0.64$, matching the predictive performance of the 108-step myopic agent. As shown in Figure 6, the 27-step non-myopic agent outperforms the 27-step myopic agent in 71% of testbed cases, the 54-step myopic agent in 61% of testbed cases, and exceeds the performance of the 108-step myopic agent in 49% of cases.

Emission rate versus source location. Normalized CRPS values for emission rate and source location (not shown in Table 2) vary substantially across strategies and flight durations. Across all conditions, normalized CRPS values for emission rate are consistently higher (i.e., worse) than those for source location. Emission rate estimation proves challenging for all strategies. For lawnmower flights with 108 steps, 36% exceeds the worst-than-prior threshold of a normalized CRPS of 1. Myopic agents show degradation rates of 34%, 43%, and 48% for 108, 54, and 27 steps, respectively. Non-myopic agents perform comparatively better, with 24%, 32%, and 38% of flights exceeding the threshold. Despite this improvement, the rates remain high, indicating difficulty in emission rate estimation. In contrast, source location estimation is notably more robust. Only 10% of lawnmower flights with 108 steps exceed the threshold. Myopic agents show 9%, 20%, and 33% for 108, 54, and 27 steps, respectively. Non-myopic agents again perform best, with lower rates of 4%, 6%, 11%. These results suggest that the source location is more reliably inferred than emission rate, particularly under shorter flight durations and with non-myopic planning.

3.3. Multi-agent framework

3.3.1. Performance statistics

Table 2 presents performance metrics for multi-agent strategies, showing that the non-myopic strategy outperforms the myopic strategy for the tested scenario. Moreover, within both strategy categories, multi-drone configurations outperform their single-drone counterparts, as illustrated in Figure 6. This suggests that parallel exploration and data collection can enhance inference quality, even without explicit coordination. Interestingly, collaborative strategies—where drones operate under a centralized policy—perform comparably to non-collaborative multi-drone setups. All reported statistics are based on sequential assimilation, where each drone's observations are processed incrementally, one after the other, at each time step.

Sequential versus *joint assimilation.* To assess the impact of the assimilation method, we also evaluated performance under joint assimilation, where observations from both agents at a given time step are assimilated simultaneously. While this approach typically yields higher cumulative information gain—

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

for example, 9.2nats compared to 7.4nats for collaborative non-myopic agents—it results in poorer predictive performance. $\mathrm{CRPS}_{\mathrm{norm}}$ increased from 0.46 under sequential assimilation to 0.55 under joint assimilation for the same test case. This trend of increased $\mathrm{CRPS}_{\mathrm{norm}}$ under joint assimilation was consistently observed across all multi-agent configurations.

3.3.2. Illustrative flights

To qualitatively demonstrate the impact of observation noise on inference in individual runs, we present selected multi-agent flight trajectories in Figure 7. The figure shows two independent drones following single-agent policies and two drones executing a centralized collaborative policy.

In both scenarios, the agents initially follow a crosswind trajectory relative to the prevailing wind before at least one agent navigates toward the inferred source location. In the non-collaborative case, apart from a noisy observation at $\boldsymbol{x} = [30,210]\,\mathrm{m}$, concentration measurements align well with the true mean concentration field. Neither drone samples near the source, yet the emission rate is inferred correctly, albeit with uncertainty. The source location is estimated reasonably well, though the agents believe it may be farther upwind than its true position. In the collaborative case, one agent obtains observations close to the source, as well as slightly upwind, and the source location is accurately inferred with high precision. Several plume observations exceed the true mean concentration field, and the emission rate is overestimated. The collaborative case achieves considerably higher cumulative information gain (9.8nats vs. 3.6nats), which is reflected in more constrained posteriors for both source location and emission rate. However, the accuracy of the inferred emission rate is worse ($\mathrm{CRPS}_Q = 0.9\,\mathrm{kg\,h^{-1}}$ vs. $0.3\,\mathrm{kg\,h^{-1}}$).

Overall performance metrics across the testbed are comparable for both non-collaborative and collaborative strategies (see Table 2). This example highlights how observation noise can degrade inference quality in specific runs and underscores the importance of sampling elevated concentrations with low noise within the true static plume. Proximity to the source enables agents to more reliably infer its

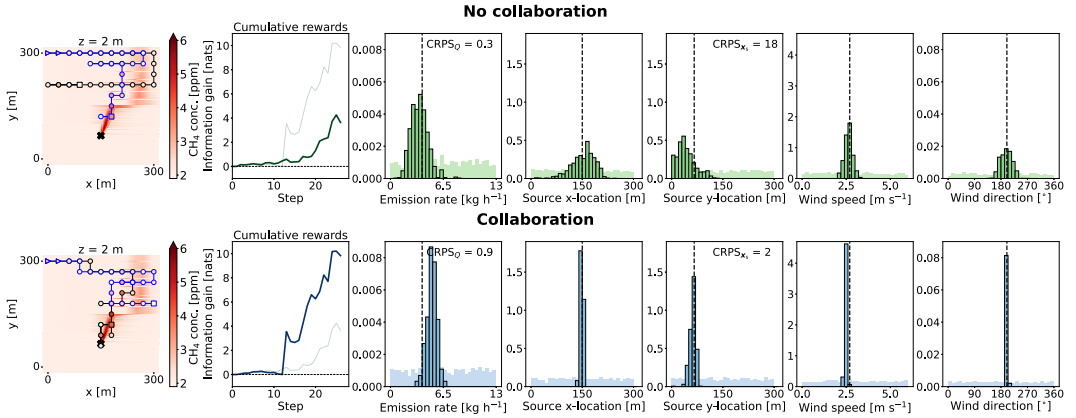


Figure 7. *Illustrative flight paths of two drones without collaboration (top), and with collaboration (bottom) using a non-myopic policy. These examples are not representative of typical behavior in either case; they were chosen to highlight the impact of observation noise. The true parameters are* $\boldsymbol{\theta} = \left[Q = 3.9\,\mathrm{kg\,h^{-1}}, x_s = 150\,\mathrm{m}, y_s = 66\,\mathrm{m}, V = 2.7\,\mathrm{m\,s^{-1}}, \Phi = 200\,^\circ\right]$*. The expected mean concentration field, following Eq. (2.3), is shown as a background heatmap, including the true source location marked by a cross. The flight paths of the two agents, shown in black and blue, are overlaid, along with the obtained noisy observations depicted by colored nodes: the first two observations are shown as a triangle, the final as a square, and intermediate observations as circles. The cumulative rewards of each flight path are plotted. The cumulative rewards of the other flight paths are shown by a thin line for comparison. Histograms display the uniform prior in light shading, the final posterior in dark shading, and the true value depicted by a dashed line.*

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

location. Visual inspection of flights across the testbed suggests that, in both collaborative and non-collaborative cases, agents consistently attempt to navigate toward the inferred source.

4. Discussion

4.1. Nature run

4.1.1. High noise level

The data-driven calibration of our observation model resulted in a challenging synthetic test case. The design reflects Arctic atmospheric conditions in a complex terrain with highly variable and turbulent wind conditions. Consequently, the simulated observations exhibit high noise levels, primarily due to turbulence, which can obscure signal patterns and introduce ambiguity into the inference process. These high-noise observations serve as a stress test for the robustness of our framework.

4.1.2. Limitations of a single test case

The nature run is based on a single scenario, with an estimated emission rate of $Q = 2.2 \pm 0.6\,\mathrm{kg\,h^{-1}}$ under specific meteorological conditions. Although we assume that the noise characteristics observed during this field experiment are broadly indicative of real-world variability, this assumption may not hold across different emission regimes or atmospheric conditions. For example, scenarios with lower emission rates or more stable wind conditions may exhibit different observational uncertainties. The selected scenario represents a specific case, characterized by highly variable wind conditions and a likely nonconstant emission rate, and may therefore not be fully representative of other field conditions.

4.1.3. Temporal correlations

The synthetic observations are generated by independently sampling from the nature run, which does not explicitly account for temporal correlations in the concentration or wind fields. However, turbulent flows are known to exhibit strong spatiotemporal correlations, meaning that the probability of encountering elevated concentrations depends on the agent's recent observation history. Recent studies using direct numerical simulations of emission plumes (Heinonen et al., 2025; Piro et al., 2025a, 2025b) have shown that when turbulent observations contain temporal correlations but the model likelihood assumes conditional independent noise, inference performance can degrade compared to scenarios where the observation model is well specified. Studies find that incorporating temporal correlations—either through correlation-aware Bayesian updates (Heinonen et al., 2025) or by leveraging an ensemble of likelihood models to mitigate misspecification (Piro et al., 2025b)—can improve the robustness and accuracy of source localization.

4.2. Single-agent framework

4.2.1. Flight paths and performance

Flight path visualizations provide insight into the observed performance statistics by revealing how agent trajectories influence information gain. Minimal-learning cases often arise when agents fail to reach the plume region, either due to suboptimal exploration or limited flight duration. Shorter flights, in particular, reduce the opportunity to find the plume and collect informative samples, thereby weakening posterior updates. In some instances, even when the plume region is reached, high observational noise prevents the agent from detecting elevated concentrations, resulting in weak posterior updates.

As shown in Figure 5, non-myopic agents are generally more successful than myopic agents at navigating toward the plume and acquiring measurements near and upwind of the source. These strategic observation locations substantially reduce uncertainty in the inferred source location. A central challenge in source inference is the ambiguity that arises when a higher emission rate from a more distant source produces concentration patterns similar to those from a lower emission rate nearby. This ambiguity is more effectively resolved when the agent obtains background concentration measurements just upwind of the source, providing information on where the source is not. Such measurements help constrain both the

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

source location and the emission rate by narrowing the plausible parameter space. Importantly, obtaining low-noise observations remains critical for reliable inference, as measurement noise can introduce bias in emission rate estimates, as illustrated in Figure 7. In contrast, myopic agents tend to remain within high-concentration regions, often failing to explore areas that would improve source localization. Lawnmower agents traverse the plume repeatedly but typically collect fewer informative samples, limiting their ability to constrain the posterior.

4.2.2. Inference challenges

The results indicate that estimating the emission rate is generally more challenging than localizing the source, particularly during shorter flight durations. This difference may partially stem from the structure of the forward model, Eq. (2.3), where concentration depends exponentially on the agent's distance to the source but linearly on the emission rate. As a result, the source location has a stronger influence on the observed signal, potentially making the emission rate harder to infer from sparse or limited observations. We identify several factors that contribute to inference challenges. First, high measurement noise can produce misleading observations that obscure the true signal. Second, recalcitrant particle degeneracy, despite rejuvenation efforts, can limit the diversity of hypotheses maintained during inference, reducing robustness. Third, equifinality can complicate parameter estimation: distinct parameter combinations can produce nearly similar measurements, making it difficult to resolve the true configuration. Moreover, we observed that particle rejuvenation may inadvertently widen the posterior when agents encounter uninformative observations, effectively diluting prior learning. This is seen, for instance, under the preplanned (lawnmower) policy.

4.3. Multi-agent framework

4.3.1. Flight paths and performance

In the tested single-source configuration, both collaborating and non-collaborating drones exhibit similar flight behavior. Agents tend to scan along the crosswind direction relative to the prevailing wind, rather than disperse. Each drone independently navigates toward regions of elevated concentration near the plume and source. This indicates that collaboration does not significantly alter the search strategy, yielding comparable performance metrics.

The main challenge lies in high observational noise, which complicates source inference. While adding a second agent improves robustness, collaboration itself offers no clear advantage—independent operation performs equally well. Clearer performance differences may emerge in more complex scenarios, such as multi-source domains, where collaboration could allow agents to divide the task and focus on distinct sources.

4.3.2. Sequential versus *joint assimilation*

Joint assimilation—processing both agents' observations simultaneously—yields higher cumulative information gain, yet consistently results in poorer predictive performance, as reflected by elevated CRPS values. This suggests that the belief distribution becomes overconfident and overly constrained, potentially centered around incorrect values. The effect likely arises from high observational noise, equifinality, and reduced particle diversity due to degeneracy. In contrast, sequential assimilation—where each observation is processed in turn with an MCMC move step—achieves lower information gain but better predictive accuracy. Gradual updates help preserve the flexibility of the belief distribution and maintain particle diversity. These findings align with the broader effectiveness of data tempering techniques in particle-based inference (Chopin and Papaspiliopoulos, 2020; Murphy, 2023).

4.3.3. Computational costs

Centralized multi-agent systems incur significantly higher computational costs compared to single-agent frameworks. Under a myopic policy, the cost scales exponentially with the number of agents, requiring

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

$\left(n_{\text{actions}} \times n_{\text{Monte Carlo steps}}\right)^{n_{\text{agents}}}$ look-ahead evaluations. This is quickly becoming infeasible in real-world applications. For non-myopic agents, policy execution remains efficient, but training time increases substantially due to the expansion of the neural network's output layer to size $n_{\text{actions}}^{n_{\text{agents}}}$. In our case, we increased the training time by a factor of 4 compared to the single-agent policy.

We note that additional training of the neural network, particularly of the multi-agent policies, may yield improved performance. Notably, the collaborative sequential assimilation policy was not trained from scratch, but retrained from the joint assimilation policy. This raises the possibility that training the sequential assimilation policy from scratch could yield different, and potentially superior, performance outcomes.

5. Lessons learned

5.1. Reality versus *simulation*

During the setup of the nature run, we found that the effective diffusivity K was particularly difficult to infer accurately. Consequently, we chose to fix its value. Furthermore, we acknowledge that wind measurements obtained from a moving drone are subject to considerable uncertainty and potential bias, particularly in the vertical component. To improve the accuracy of effective diffusivity estimates, it may be beneficial to install a stationary sonic anemometer near the study site (van Hove et al., 2025). Although limited to a single location, this would likely provide more stable and reliable wind measurements, thereby supporting more accurate modeling of atmospheric dispersion.

5.2. Entropy computation

Accurate entropy computation, based on the particle distribution, proved essential for the success of our framework. Earlier attempts using Eq. (2.16) were unsuccessful, likely due to its inability to capture the true uncertainty represented by the particle ensemble. While the current method yields more reliable results, it is computationally expensive. A potential alternative is to compute entropy discretely by binning particles across the parameter space and estimating entropy from particle counts per bin. This approach could reduce computational costs, though it introduces a trade-off between precision and efficiency. The number of bins must be chosen carefully: more bins improve resolution but increase computational cost, while fewer bins risk oversimplifying the belief structure.

5.3. Balancing information gain and belief accuracy

Using information gain as a reward signal inherently encourages particle degeneracy, as it favors concentrated belief distributions. Rejuvenation methods help mitigate the risk of converging on incorrect values. Achieving a balance between belief constraining and avoiding overconfidence requires careful tuning of the computational implementation of the Bayesian inference process. Key factors include the number of particles, the number and step size of MCMC moves, and the choice between sequential and joint assimilation. These must be managed within the constrained computational budget required for both training and deployment, where fast decision-making is essential. Parameters that perform well in one setting may lead to poor generalization in another.

In our implementation, we adopted an adaptive MCMC step size based on the shape of the dynamic prior: larger steps for wide priors and smaller steps for narrow distributions. This allowed us to reduce the particle count to just 1,000 while maintaining reasonable inference quality. However, we observed belief widening in response to uninformative observations, highlighting the sensitivity of the system to observation quality.

5.4. Belief state representation

Representation of the belief state within the neural network was refined through trial and error. The final input configuration, detailed in Appendix E, includes the belief over source location and wind direction,

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

the drone's current position, and a visitation map that tracks how frequently each cell in the domain has been visited. We also experimented with including the drone's battery level instead of the visitation map, but this proved less effective. While battery level provides a sense of remaining operational time, it does not convey spatial coverage. Without the inclusion of the visitation map, the agent tended to remain in grid cells with high concentration, neglecting areas at or upwind of the source, which are often helpful for accurate localization. We also tested including belief over emission rate and wind speed in the form of mean raw normalized values. However, these additions did not considerably improve cumulative information gain. As a result, they were excluded from the state representation.

6. Future work

6.1. More realism in synthetic simulations

We used a nature run to create more realistic simulations, but several opportunities remain to enhance realism. First, constructing additional nature runs across varied environments and emission regimes, such as different source strengths and atmospheric conditions, would help refine the likelihood model and assess its generalizability. Second, the forward model could be extended to include more physical processes (Pirk et al., 2022), such as vertical wind profiles, diffusivity anisotropy, and temporal correlations inferred via observations or large-eddy simulations, to more realistically simulate the spatiotemporal structure of plume dispersion. Surrogate or reduced-order models may offer a computationally efficient way to incorporate such complexity (e.g., Gunawardena et al., 2021; Lumet et al., 2025).

6.2. Navigational flexibility

In the current framework, the source location is modeled in a continuous spatial domain, while the drone operates on a discretized grid. This mismatch limits the flexibility of path planning. Transitioning to a continuous domain would allow agents to select both heading direction and step size, enabling larger exploratory movements early in the flight and more precise steps as the plume is detected. Such flexibility would better reflect real-world drone capabilities and support more adaptive and efficient trajectories. Standard Q-learning is not well suited for continuous action spaces. Park et al. (2022) used a Deep Deterministic Policy Gradient algorithm to train an AIPP policy for the STE problem with a continuous state space and an action space defined by heading direction and fixed step size.

Future extensions could also introduce structured maneuvers into the action space, such as curtain flights. Instead of always directing the drone to a single point, the policy could optionally select a vertical curtain sweep at a chosen downwind distance from the estimated source location or a horizontal sweep at a selected altitude. This hybrid action space could make the problem more tractable, as it may reduce the number of individual action-selection steps that need to be trained. Notably, the trained non-myopic policy already demonstrates this behavior: the drone typically begins its trajectory with a horizontal crosswind sweep positioned downwind of the source.

6.3. Multi-agent systems

Although our test case did not demonstrate a clear performance advantage for collaborative over non-collaborative strategies, this outcome may be specific to the scenario studied and the methods used. Future work should explore configurations where the benefits of coordination may become more pronounced, such as domains with multiple emission sources, as well as different AIPP strategies, inference algorithms, and belief state representations used as input to the neural network. Notably, Heinonen et al. (2025) suggests that multiple collaborating agents may be able to exploit temporal correlations in the flow. Future work can focus on extending the multi-agent framework to incorporate error correlation-aware Bayesian inference updates and memory-based RL policies. Decentralized systems—where drones collaborate and share information but make decisions independently—may offer better scalability and practical feasibility than centralized collaboration. Prior studies, including Park and Oh (2020) and Ristic

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

et al. (2020), investigated the performance of decentralized myopic multi-agent systems for the STE problem.

6.4. Real-world experiments

Conducting real-world experiments remains the ultimate goal for validating the proposed framework. Field deployments would enable assessment under true environmental variability, offering insights into robustness and operational feasibility. A central challenge in RL is developing agents that generalize across diverse environments. Atmospheric and terrain conditions can vary significantly between deployment sites, influencing plume behavior. Future work should, therefore, focus on developing a framework capable of adapting to a wide range of conditions, including different meteorological regimes, terrain types, source characteristics, and ideally, varying number of emission sources.

7. Conclusion

This study presents a framework for methane STE using drone-based observations and adaptive informative path planning strategies. By fusing Bayesian inference with reinforcement learning, we demonstrate that adaptive agents (i.e., drones) can efficiently maximize information gain and improve the accuracy of emission rate and source location estimates. Our approach is distinguished by its use of a nature run with a calibrated observation model derived from real-world data, providing a more representative basis for synthetic experiments that capture realistic atmospheric variability. The challenging conditions of the field experiment introduced substantial observational noise, serving as a stress test for the robustness of various flight strategies and durations.

Among the tested strategies, non-myopic policies trained via deep reinforcement learning consistently outperformed both myopic and preplanned (lawnmower) approaches across multiple metrics, including entropy reduction and proper scoring rules. These agents demonstrated superior navigation toward informative regions, often reaching the inferred source location and sampling upwind. The non-myopic agents maintained robust performance even for shorter flight durations.

We also investigated multi-agent collaboration, finding comparable performance between collaborative and non-collaborative strategies in the tested single-source scenario. Performance differences may become more pronounced in multi-source domains where collaboration could allow agents to divide the task and focus on distinct sources. Sequential assimilation interlaced with rejuvenation offered a more balanced trade-off between belief accuracy and diversity than joint assimilation, which tended to produce overconfident and less reliable estimates due to particle degeneracy.

Three components were critical to the success of this framework: (i) accurate entropy computation based on the full particle distribution, rather than particle weights alone, (ii) careful tuning of the computational implementation of the particle-based Bayesian inference process to balance belief constraint with rejuvenation, while maintaining computational efficiency, and (iii) refinement of the belief state representation in the reinforcement learning process to ensure that the agent receives informative and spatially relevant input features.

Overall, our findings support the development of probabilistic machine learning methods for intelligent, tractable, and data-driven active inference strategies for drone-based environmental monitoring. By improving the precision, accuracy, efficiency, and reliability of methane source term estimates, these methods can contribute to more accurate emission inventories and inform targeted climate mitigation efforts.

Open peer review. To view the open peer review materials for this article, please visit http://doi.org/10.1017/eds.2026.10029.

Acknowledgments. The authors would like to thank Store Norske, and in particular Malte Jochmann, for their contribution to the field experiment. Furthermore, the authors would like to thank Marius Jonassen for his suggestion to take measurements at the borehole. The authors would also like to thank Gina Schulz and Marcos Porcires for their assistance in the field.

AI tools were used in the writing process of this manuscript to enhance clarity and coherence of the written language. The tools were not used to generate original ideas or content. All AI-assisted output was evaluated, edited, and verified by the authors.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

Author contribution. Conceptualization: A.v.H., K.A. and N.P. Methodology: A.v.H., K.A., and N.P. Investigation: A.v.H. Formal analysis: A.v.H., K.A., and N.P. Software: A.v.H. Data visualization: A.v.H. Writing original draft: A.v.H. Writing review and editing: A.v.H., K.A., and N.P. Funding acquisition: N.P. All authors approved the final submitted draft.

Competing interests. The authors declare none.

Data availability statement. Our code is publicly available on GitHub at: https://github.com/AlouetteUiO/active.

Ethics statement. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

Funding statement. This work was supported by the Research Council of Norway (projects #333232 (CircAgric-GHG) and #301552 (Spot-On)) and the European Research Council (project #101116083 (ACTIVATE)). This work is a contribution to the strategic research initiative LATICE (#UiO/GEO103920), the Center for Biogeochemistry in the Anthropocene, as well as the Center for Computational and Data Science at the University of Oslo.

References

Albrecht SV, **Christianos F and Schäfer L** (2024) *Multi-Agent Reinforcement Learning: Foundations and Modern Approaches*. Cambridge, Massachusetts: The MIT Press

Birchall T, **Jochmann M**, **Betlem P**, **Senger K**, **Hodson A and Olaussen S** (2023) Permafrost trapped natural gas in Svalbard, Norway. *Frontiers in Earth Science 11*, 1277027. https://doi.org/10.3389/feart.2023.1277027.

Boers Y, **Driessen H**, **Bagchi A and Mandal P** (2010) Particle filter based entropy. In *13th International Conference on Information Fusion*. Edinburgh: IEEE, pp. 1–8. https://doi.org/10.1109/ICIF.2010.5712013.

Box GEP and Cox DR (1964) An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological) 26*(2), 211–252.

Chopin N (2002) A sequential particle filter method for static models. *Biometrika 89*, 539–552. https://doi.org/10.1093/biomet/89.3.539.

Chopin N and Papaspiliopoulos O (2020) *An Introduction to Sequential Monte Carlo*, 1st Edn. Cham: Springer Nature

Kingma DP and Ba LJ (2015) Adam: A Method for Stochastic Optimization. In *International Conference for Learning Representations*. arXiv:1412.6980[cs.LG]. https://doi.org/10.48550/arXiv.1412.6980.

Doucet A and Johansen AM (2009) A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering 12*, 656–705. https://doi.org/10.48550/arXiv.1412.6980.

Fischer J and Tas ÖS (2020) Information particle filter tree: An online algorithm for POMDPs with belief-based rewards on continuous domains. In III, Hal Daumé and Singh Aarti (eds), *Proceedings of the 37th International Conference on Machine Learning*, Vol. *119*. PMLR, pp. 3177–3187.

Francis A, **Li S**, **Griffiths C and Sienz J** (2022) Gas source localization and mapping with mobile robots: A review. *Journal of Field Robotics 39*, 1341–1373. https://doi.org/10.1002/rob.22109.

Gilks WR and Berzuini C (2001) Following a moving target—Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society Series B: Statistical Methodology 63*(1), 127–146. https://doi.org/10.1111/1467-9868.00280.

Gisbert FJG (2003) Weighted samples, kernel density estimators and convergence. *Empirical Economics 28*(2), 335–351. https://doi.org/10.1007/s001810200134.

Gneiting T and Raftery AE (2007) Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association 102*(477), 359–378. https://doi.org/10.1198/016214506000001437.

Gunawardena N, **Pallotta G**, **Simpson M and Lucas DD** (2021) Machine learning emulation of spatial deposition from a multi-physics Ensemble of Weather and Atmospheric Transport Models. *Atmosphere 12*(8), 953. https://doi.org/10.3390/atmos12080953.

Heinonen RA, **Biferale L**, **Celani A and Vergassola M** (2023) Optimal policies for Bayesian olfactory search in turbulent flows. *Physical Review E 107*(5), 055105. https://doi.org/10.1103/PhysRevE.107.055105.

Heinonen RA, **Biferale L**, **Celani A and Vergassola M** (2025) Exploring Bayesian olfactory search in realistic turbulent flows. *Physical Review Fluids 10*(6), 064614. https://doi.org/10.1103/9q6q-nlxc.

Hersbach H (2000) Decomposition of the continuous ranked probability score for ensemble prediction systems. *Weather and Forecasting 15*(5), 559–570. https://doi.org/10.1175/1520-0434(2000)015<0559:DOTCRP>2.0.CO;2.

van Hove A, **Aalstad K**, **Lind V**, **Arndt C**, **Odongo V**, **Ceriani R**, **Fava F**, **Hulth J and Pirk N** (2025) Inferring methane emissions from African livestock by fusing drone, tower, and satellite data. *Biogeosciences 22*(16), 4163–4186. https://doi.org/10.5194/bg-22-4163-2025.

van Hove A, **Aalstad K and Pirk N** (2023) Using reinforcement learning to improve drone-based inference of greenhouse gas fluxes. *Nordic Machine Intelligence 3*, 1–6. https://doi.org/10.5617/nmi.9897.

van Hove A, **Aalstad K and Pirk N** (2024) Guiding drones by information gain. In Lutchyn T, Ramírez Rivera A and Ricaud B (eds), *Proceedings of the 5th Northern Lights Deep Learning Conference (NLDL)*, Vol. *233*. PMLR, pp. 89–96.

Hutchinson M, **Liu C and Chen WH** (2019a) Information-based search for an atmospheric release using a Mobile robot: Algorithm and experiments. *IEEE Transactions on Control Systems Technology 27*(6), 2388–2402. https://doi.org/10.1109/TCST.2018.2860548.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

Hutchinson M, **Liu C and Chen WH** (2019b) Source term estimation of a hazardous airborne release using an unmanned aerial vehicle. *Journal of Field Robotics 36*(4), 797–817. https://doi.org/10.1002/rob.21844.

Hutchinson M, **Liu C**, **Thomas P and Chen WH** (2020) Unmanned aerial vehicle-based hazardous materials response: Information-theoretic hazardous source search and reconstruction. *IEEE Robotics & Automation Magazine 27*(3), 108–119. https://doi.org/10.1109/MRA.2019.2943006.

Hutchinson M, **Oh H and Chen WH** (2017) A review of source term estimation methods for atmospheric dispersion events using static or mobile sensors. *Information Fusion 36*, 130–148. https://doi.org/10.1016/j.inffus.2016.11.010.

IPCC (2023) Summary for policymakers. In Lee H and Romero J (eds), *Climate Change 2023: Synthesis Report. Contribution of Working Groups I, II and III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Geneva: IPCC, pp. 1–34. https://doi.org/10.59327/IPCC/AR6978929169164 7.001

Jaynes E (2003) *Probability Theory: The Logic of Science*. Cambridge, England: Cambridge University Press. https://doi.org/10.1017/CBO9780511790423

Kochenderfer MJ, **Wheeler TA and Wray KH** (2022) *Algorithms for Decision Making*. Cambridge, Massachusetts: The MIT Press

Lee J, **Jang H**, **Park M and Oh H** (2025) Enhanced reward function Design for Source Term Estimation Based on deep reinforcement learning. *IEEE Access 13*, 87777–87792. https://doi.org/10.1109/ACCESS.2025.3569827.

van Leeuwen P (2015) Representation errors and retrievals in linear and nonlinear data assimilation. *Quarterly Journal of the Royal Meteorological Society 141*, 1612–1623. https://doi.org/10.1002/qj.2464.

Lindley DV (1956) On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics 27*(4), 986–1005. https://doi.org/10.1214/aoms/1177728069.

Loisy A and Eloy C (2022) Searching for a source without gradients: How good is infotaxis and how to beat it. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 478*(2262), 20220118. https://doi.org/10.1098/rspa.2022.0118.

Loisy A and Heinonen RA (2023) Deep reinforcement learning for the olfactory search POMDP: A quantitative benchmark. *The European Physical Journal E 46*(3), 17. https://doi.org/10.1140/epje/s10189-023-00277-8.

Lumet E, **Rochoux MC**, **Jaravel T and Lacroix S** (2025) Uncertainty-aware surrogate modeling for urban air pollutant dispersion prediction. *Building and Environment 267*, 112287. https://doi.org/10.1016/j.buildenv.2024.112287.

Lunt MF, **Harris SJ**, **Hacker J**, **Joynes I**, **Robertson T**, **Thompson S and France JL** (2025) An evaluation of airborne mass balance and tracer correlation approaches to estimate site-level CH_4 emissions from LNG facilities using CO_2 as a tracer of opportunity. *Atmospheric Measurement Techniques 18*(17), 4413–4431. https://doi.org/10.5194/amt-18-4413-2025.

MacKay DJC (2003) *Information Theory, Inference, and Learning Algorithms*. Cambridge, England: Cambridge University Press

Martín A, **Agarwal A**, **Barham P**, **Brevdo E**, **Chen Z**, **Citro C**, **Corrado GS**, **Davis A**, **Dean J**, **Devin M**, **Ghemawat S**, **Goodfellow I**, **Harp A**, **Irving G**, **Isard M**, **Jia Y**, **Jozefowicz R**, **Kaiser L**, **Kudlur M**, **Levenberg J**, **Mané D**, **Monga R**, **Moore S**, **Murray D**, **Olah C**, **Schuster M**, **Shlens J**, **Steiner B**, **Sutskever I**, **Talwar K**, **Tucker P**, **Vanhoucke V**, **Vasudevan V**, **Viégas F**, **Vinyals O**, **Warden P**, **Wattenberg M**, **Wicke M**, **Yu Y and Zheng X** (2015) TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Google Research. Software available at: https://www.tensorflow.org/.

Masutani M, **Schlatter TW**, **Errico RM**, **Stoffelen A**, **Andersson E**, **Lahoz W**, **Woollen JS**, **Emmitt GD**, **Riishøjgaard LP and SJ** (2010) Observing system simulation experiments. In Lahoz W, Khattatov B and Menard R (eds), *Data Assimilation: Making Sense of Observations*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 647–679. https://doi.org/10.1007/978-3-540-74703-1_24

Mnih V, **Kavukcuoglu K**, **Silver D**, **Rusu AA**, **Veness J**, **Bellemare MG**, **Graves A**, **Riedmiller M**, **Fidjeland AK**, **Ostrovski G**, **Petersen S**, **Beattie C**, **Sadik A**, **Antonoglou I**, **King H**, **Kumaran D**, **Wierstra D**, **Legg S and Hassabis D** (2015) Human-level control through deep reinforcement learning. *Nature 518*, 529–533. https://doi.org/10.1038/nature14236.

Murphy KP (2022) *Probabilistic Machine Learning: An Introduction. Adaptive Computation and Machine Learning Series*. Cambridge: The MIT Press, pp. 826

Murphy KP (2023) *Probabilistic Machine Learning: Advanced Topics. Adaptive Computation and Machine Learning Series*. Cambridge: The MIT Press, pp. 1

Odland T (2018) KDEpy: Kernel Density Estimation in Python (Version 0.9.10) [Software] Zenodo. https://doi.org/10.5281/zenodo.2392268.

Park M, **An S**, **Seo J and Oh H** (2021) Autonomous source search for UAVs using Gaussian mixture model-based Infotaxis: Algorithm and flight experiments. *IEEE Transactions on Aerospace and Electronic Systems 57*(6), 4238–4254. https://doi.org/10.1109/TAES.2021.3098132.

Park M, **Ladosz P and Oh H** (2022) Source term estimation using deep reinforcement learning with Gaussian mixture model feature extraction for Mobile sensors. *IEEE Robotics and Automation Letters 7*(3), 8323–8330. https://doi.org/10.1109/LRA.2022.3184787.

Park M and Oh H (2020) Cooperative information-driven source search and estimation for multiple agents. *Information Fusion 54*, 72–84. https://doi.org/10.1016/j.inffus.2019.07.007.

Pirk N, **Aalstad K**, **Westermann S**, **Vatne A**, **van Hove A**, **Tallaksen LM**, **Cassiani M and Katul G** (2022) Inferring surface energy fluxes using drone data assimilation in large eddy simulations. *Atmospheric Measurement Techniques 15*, 7293–7314. https://doi.org/10.5194/amt-15-7293-2022.

Piro L, **Heinonen RA**, **Carbone M**, **Biferale L and Cencini M** (2025a) Policy heterogeneity improves collective olfactory search in 3-D turbulence. *arXiv:2504.11291[physics.flu-dyn]*. https://doi.org/10.48550/arXiv.2504.11291.

Piro L, **Heinonen RA**, **Cencini M and Biferale L** (2025b) Many wrong models approach to localise an odour source in turbulence with static sensors. *Journal of Turbulence 26*(5), 153–173. https://doi.org/10.1080/14685248.2025.2492711.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

Popović M, Ott J, Rückin J and Kochenderfer MJ (2024) Learning-based methods for adaptive informative path planning. *Robotics and Autonomous Systems 179*, 104727. https://doi.org/10.1016/j.robot.2024.104727.

Prather MJ, Holmes CD and Hsu J (2012) Reactive greenhouse gas scenarios: Systematic exploration of uncertainties and the role of atmospheric chemistry. *Geophysical Research Letters 39*(9), L09803. https://doi.org/10.1029/2012gl051440.

Ristic B, Gilliam C, Moran W and Palmer JL (2020) Decentralised multi-platform search for a hazardous source in a turbulent flow. *Information Fusion 58*, 13–23. https://doi.org/10.1016/j.inffus.2019.12.011.

Sanz-Alonso D, Stuart A and Taeb A (2023) *Inverse Problems and Data Assimilation*, 1st Edn. Vol. *107*. Cambridge, England: Cambridge University Press.

Särkkä S and Svensson L (2023) *Bayesian Filtering and Smoothing*, 2nd Edn. Cambridge, England: Cambridge University Press. https://doi.org/10.1017/9781108917407

Saunois M, Martinez A, Poulter B, Zhang Z, Raymond P, Regnier P, Canadell JG, Jackson RB, Patra PK, Bousquet P, Ciais P, Dlugokencky EJ, Lan X, Allen GH, Bastviken D, Beerling DJ, Belikov DA, Blake DR, Castaldi S, Crippa M, Deemer BR, Dennison F, Etiope G, Gedney N, Höglund-Isaksson L, Holgerson MA, Hopcroft PO, Hugelius G, Ito A, Jain AK, Janardanan R, Johnson MS, Kleinen T, Krummel P, Lauerwald R, Li T, Liu X, McDonald KC, Melton JR, Mühle J, Müller J, Murguia-Flores F, Niwa Y, Noce S, Pan S, Parker RJ, Peng C, Ramonet M, Riley WJ, Rocher-Ros G, Rosentreter JA, Sasakawa M, Segers A, Smith SJ, Stanley EH, Thanwerdas J, Tian H, Tsuruta A, Tubiello FN, Weber TS, Van Der Werf G, Worthy DE, Xi Y, Yoshida Y, Zhang W, Zheng B, Zhu Q, Zhu Q and Zhuang Q (2024) Global methane budget 2000–2020. *Earth System Science Data 17*(5), 1873–1958. https://doi.org/10.5194/essd-2024-115.

Saunois M, Stavert AR, Poulter B, Bousquet P, Canadell JG, Jackson RB, Raymond PA, Dlugokencky EJ, Houweling S, Patra PK, Ciais P, Arora VK, Bastviken D, Bergamaschi P, Blake DR, Brailsford G, Bruhwiler L, Carlson KM, Carrol M, Castaldi S, Chandra N, Crevoisier C, Crill PM, Covey K, Curry CL, Etiope G, Frankenberg C, Gedney N, Hegglin MI, Höglund-Isaksson L, Hugelius G, Ishizawa M, Ito A, Janssens-Maenhout G, Jensen KM, Joos F, Kleinen T, Krummel PB, Langenfelds RL, Laruelle GG, Liu L, Machida T, Maksyutov S, McDonald KC, McNorton J, Miller PA, Melton JR, Morino I, Müller J, Murguia-Flores F, Naik V, Niwa Y, Noce S, O'Doherty S, Parker RJ, Peng C, Peng S, Peters GP, Prigent C, Prinn R, Ramonet M, Regnier P, Riley WJ, Rosentreter JA, Segers A, Simpson IJ, Shi H, Smith SJ, Steele LP, Thornton BF, Tian H, Tohjima Y, Tubiello FN, Tsuruta A, Viovy N, Voulgarakis A, Weber TS, Van Weele M, Van Der Werf GR, Weiss RF, Worthy D, Wunch D, Yin Y, Yoshida Y, Zhang W, Zhang Z, Zhao Y, Zheng B, Zhu Q, Zhu Q and Zhuang Q (2020) The global methane budget 2000–2017. *Earth System Science Data 12*(3), 1561–1623. https://doi.org/10.5194/essd-12-1561-2020.

Scheutz C, Knudsen J, Vechi N and Knudsen J (2025) Validation and demonstration of a drone-based method for quantifying fugitive methane emissions. *Journal of Environmental Management 373*, 123467. https://doi.org/10.1016/j.jenvman.2024.123467.

Smith T, Sharma A, Marshall L, Mehrotra R and Sisson S (2010) Development of a formal likelihood function for improved Bayesian inference of ephemeral catchments. *Water Resources Research 46*(12), W12551. https://doi.org/10.1029/2010WR009514.

Stull RB (1989) *An Introduction to Boundary Layer Meteorology*. 2nd Edn. reprint. Dordrecht: Kluwer Academic Publishers.

Sutton RS and Barto AG (2018) *Reinforcement Learning: An Introduction*. 2nd Edn. Cambridge, Massachusetts: The MIT Press

Vergassola M, Villermaux E and Shraiman BI (2007) 'Infotaxis' as a strategy for searching without gradients. *Nature 445*, 406–409. https://doi.org/10.1038/nature05464.

A. Businger–Dyer relationship

To estimate the effective diffusivity K, we apply the Businger–Dyer relationship (Stull, 1989), a widely used parameterization for turbulent transport in the atmospheric surface layer, to real-world data collected during the field experiment.

We estimate the friction velocity u_* using sonic anemometer measurements at three flight altitudes: 2, 4, and 6 m. Although u_* is not altitude-specific, we compute separate estimates from each dataset to account for variability in the wind measurements across altitudes:

$$u_* = \left(\overline{u'w'}^2 + \overline{v'w'}^2\right)^{1/4}, \tag{A.1}$$

where u', v', and w' are the turbulent fluctuations in the horizontal and vertical wind components, and the overbars denote time-averaged covariances.

The Obukhov length L is then calculated as:

$$L = -\frac{\overline{\theta}_v u_*^3}{\kappa g \overline{w'\theta'_v}}, \tag{A.2}$$

where $\overline{\theta}_v$ is the virtual potential temperature, $\kappa = 0.4$ is the von Kármán constant, and $g = 9.81\ \mathrm{m\,s^{-2}}$ is the gravitational acceleration. Due to limited humidity data, we approximate $\overline{\theta}_v$ using potential temperature $\overline{\theta}_v \approx \overline{\theta}$. A small to moderately positive Obukhov length indicates stable atmospheric stratification, which was observed for the field experiment.

Under stable conditions, the vertical diffusivity profile $K(z)$ is given by:

$$K(z) = \frac{\kappa z u_*}{\Psi_H}, \tag{A.3}$$

where Ψ_H is the stability correction function for heat, defined as:

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

$$\Psi_H = 0.74 + 4.7\frac{z}{L}. \tag{A.4}$$

We evaluate $K(z)$ at a reference height of $z = 4\,\mathrm{m}$ using measurements from all three altitudes, yielding estimates of 0.7, 1.0, and 1.3 $\mathrm{m^2\ s^{-1}}$, respectively. We adopt the average of these values as the effective diffusivity used in our synthetic experiments: $K = 1.0\,\mathrm{m^2\,s^{-1}}$.

We acknowledge that wind measurements from a moving drone are subject to considerable uncertainty and potential bias, particularly in the vertical component. As such, this method provides only a rough estimate. For the purposes of our synthetic experiments, the precise value of K is less critical than ensuring methodological consistency across all tested configurations.

B. Box–Cox transformation and MLE

In our framework, wind speed and direction are modeled as static global parameters, independent of location. For example, wind speed is expressed as $v(\boldsymbol{x},t) = V + \epsilon_V$. This simplicity allows us to treat the multiple noisy measurements from the field experiment as repeated observations of the same latent variable. Under this assumption, we estimate the Box–Cox transformation parameters via MLE (Kochenderfer et al., 2022; Murphy, 2022), assuming approximate normality in the transformed space.

The Box–Cox transformation (Box and Cox, 1964) maps strictly positive, non-normal data to a distribution that more closely approximates normality. For a data point d_i (e.g., wind speed or direction), it is defined as:

$$d_i^{(\lambda)} = \begin{cases} \dfrac{d_i^{\lambda} - 1}{\lambda} & \text{if} \lambda \neq 0, \\ \ln(d_i) & \text{if} \lambda = 0, \end{cases} \tag{B.1}$$

This family includes several commonly used cases: $\lambda = 1$ (identity), $\lambda = 0$ (logarithmic), $\lambda = 0.5$ (square root), and $\lambda = -1$ (reciprocal).

We apply the transformation to wind speed measurements from the field experiment. The transformed variables are modeled as approximately Gaussian:

$$\begin{aligned} v(\boldsymbol{x},t)^{(\lambda)} &\sim \mathcal{N}(\mu_v^{(\lambda)}, \sigma_v^{(\lambda)}), \\ &\approx \mu_v^{(\lambda)} + \varepsilon_v^{(\lambda)}, \text{ where } \varepsilon_v^{(\lambda)} \sim \mathcal{N}(0, \sigma_v^{(\lambda)}). \end{aligned} \tag{B.2}$$

Since the true wind field and emission parameters of the field experiment are unknown, residuals cannot be computed directly. Instead, we estimate the transformation parameters directly using wind measurements, assuming approximate normality in the transformed space. These estimated parameters are then used to define the observation (including representation) error model—specifically, the assumed distribution of residuals in the transformed space. The transformed likelihood function, Eq. (2.7), used in the synthetic experiments is given by:

$$p\left(\boldsymbol{d}^{(\lambda)} | \boldsymbol{\theta}\right) = \det(2\pi \mathbf{R}_\lambda)^{-1/2} \exp\left(-\frac{1}{2}\left[\boldsymbol{d}^{(\lambda)} - \widehat{\boldsymbol{d}}^{(\lambda)}\right] \mathbf{R}_\lambda^{-1} \left[\boldsymbol{d}^{(\lambda)} - \widehat{\boldsymbol{d}}^{(\lambda)}\right]\right), \tag{B.3}$$

where $\mathbf{R}_\lambda$ is the observation error covariance matrix in the transformed space, with diagonal entries containing the squared values of $\sigma^{(\lambda)}$. The transformed model-predicted observations are defined as:

$$\widehat{\boldsymbol{d}}^{(\lambda)} = \mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\zeta})^{(\lambda)}. \tag{B.4}$$

The optimal transformation parameter λ_* for wind speed is estimated via MLE, assuming independent and identically distributed measurements. We maximize the likelihood of the transformed measurements under a Gaussian model:

$$\lambda_* = \arg\max_{\lambda} \sum_i \log p\left(v_i^{(\lambda)} | \mu_v^{(\lambda)}, \sigma_v^{(\lambda)}\right), \tag{B.5}$$

where λ and $\sigma_v^{(\lambda)}$ are estimated from the transformed data for each candidate λ.

We assume homoscedasticity in the transformed space—that is, constant variance of the observation error as a function of wind speed. This allows us to treat the transformed wind speed observations as samples from a Gaussian distribution, enabling direct application of MLE. In the original (untransformed) space, however, the error model becomes heteroscedastic, especially for $\lambda < 1$.

Methane concentration is modeled as a spatially varying field, that is, $c(\boldsymbol{x},t) = C(\boldsymbol{x}) + \epsilon_C$, where $C(\boldsymbol{x})$ denotes the latent, temporally averaged concentration at each location. These latent variables are indirectly related to the uncertain parameters $\boldsymbol{\theta}$ through a nonlinear forward model. Due to the lack of repeated observations per location and the nonlinearity of the mapping, direct MLE is not applicable. Instead, we use Bayesian model comparison to select appropriate transformation parameters, evaluating competing models based on their evidence (see Appendix C).

C. Bayesian model comparison

To calibrate the Box–Cox transformation applied to the concentration observation model in the nature run, we use Bayesian model comparison to select the transformation parameters λ and $\sigma^{(\lambda)}$ that best explains the field experiment data.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

Bayes' rule, written to emphasize the dependence on the observation model $\mathcal{M}$, is given by:

$$p(\boldsymbol{\theta}|\boldsymbol{d},\mathcal{M}) = \frac{p(\boldsymbol{d}|\boldsymbol{\theta},\mathcal{M})p(\boldsymbol{\theta}|\mathcal{M})}{p(\boldsymbol{d}|\mathcal{M})}, \tag{C.1}$$

where the denominator $p(\boldsymbol{d}|\mathcal{M})$ is the model evidence. This term quantifies how well the observation model $\mathcal{M}$, including the transformation parameters, explains the observed data.

To isolate the effect of the Box–Cox transformation, we fix all other structural components of the observation model and vary only the transformation parameters λ and $\sigma^{(\lambda)}$ applied to the concentration data. For the model comparison, we assume known source coordinates $\boldsymbol{x}_s = [x_s, y_s]$ and infer only the emission rate Q and nuisance parameters: mean wind speed V and mean wind direction Φ.

Direct computation of the model evidence is challenging in our setting due to the use of particle filters with MCMC rejuvenation, which disrupts the tractable relationship between the prior and posterior distribution. To address this, we use importance sampling with a proposal distribution $q(\boldsymbol{\theta})$. To approximate the posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{d},\mathcal{M})$, we draw samples from the proposal distribution $\boldsymbol{\theta}^{(j)} \sim q(\boldsymbol{\theta})$ and assign each sample a normalized importance weight:

$$W^{(j)} = \frac{\overline{W}^{(j)}}{\sum_{k=1}^{N_e} \overline{W}^{(k)}}, \text{ where } \overline{W}^{(j)} = \frac{p(\boldsymbol{d}|\boldsymbol{\theta}^{(j)},\mathcal{M})p(\boldsymbol{\theta}^{(j)}|\mathcal{M})}{q(\boldsymbol{\theta}^{(j)})}. \tag{C.2}$$

This yields a weighted set of particles $\left\{\left(\boldsymbol{\theta}^{(j)}, W^{(j)}\right)\right\}_{j=1}^{N_e}$ that approximates the posterior.

This is the standard approach to self-normalized importance sampling (Chopin and Papaspiliopoulos, 2020; Murphy, 2023), commonly used in particle filtering methods (Särkkä and Svensson, 2023). It is noteworthy that we use W for importance sampling weights to distinguish them from the weights w in Section 2.1.3 to reflect the use of a different proposal distribution.

Importantly, this importance sampling scheme also yields an estimate of the model evidence:

$$p(\boldsymbol{d}|\mathcal{M}) \simeq \frac{1}{N_e}\sum_{j=1}^{N_e} \overline{W}^{(j)}, \tag{C.3}$$

which is used implicitly in the self-normalization of the weights.

We employ the particle-based approximation of model evidence, Eq. (C.3), for Bayesian model comparison of the Box–Cox parameters in the concentration observation model $\mathcal{M}$. Since we are interested in the model evidence for the untransformed observations $p(\boldsymbol{d}|\mathcal{M})$, rather than for the Box–Cox transformed data $p\left(\boldsymbol{d}^{(\lambda)}|\mathcal{M}\right)$, we apply a correction involving the Jacobian of the Box–Cox transformation. This correction is introduced after evaluating the transformed evidence using the transformed likelihood, Eq. (B.3).

The proposal distribution $q(\boldsymbol{\theta})$ is constructed from the posterior obtained via Bayesian inference using a particle filter. For the nature run, data processing is conducted post-collection, allowing for inference with $N_e = 500{,}000$ particles and five MCMC steps per update. Observations are assimilated in mini-batches of 30 sets of $[c(\boldsymbol{x},t), v(\boldsymbol{x},t), \phi(\boldsymbol{x},t)]$. The resulting posterior is then approximated by a truncated Gaussian, serving as the proposal distribution for batch smoothing via importance sampling. In this context, batch smoothing refers to assimilating all available observations in a single step, rather than sequentially as in filtering. This approach omits the need for MCMC rejuvenation and yields a globally informed posterior over the parameters $\boldsymbol{\theta}$, which is critical for robust model evidence estimation.

We evaluate the model evidence $p(\boldsymbol{d}|\mathcal{M})$ across a range of candidate values for the transformation parameters λ and $\sigma^{(\lambda)}$, and identify the combination that maximizes this quantity. The selected parameters are then incorporated into the Box–Cox transformed likelihood function, Eq. (B.3), used in the transformed observation space for the synthetic experiments.

D. Kernel Density Estimation

To compute the entropy of the posterior distribution over the source term parameters, we employ Kernel Density Estimation (KDE) (Gisbert, 2003) to approximate the probability density at each particle location. This enables a smooth, continuous estimate of the posterior, which we use to evaluate Eq. (2.15). Kernel density estimates were computed in Python using the KDEpy library (Odland, 2018).

Given the ensemble of weighted particles $\left\{\left(\boldsymbol{\theta}^{(j)}, w^{(j)}\right)\right\}_{j=1}^{N_e}$, Eq. (2.8), the KDE at the location of the particle $\boldsymbol{\theta}^{(j)}$ is defined as:

$$\widehat{p}\left(\boldsymbol{\theta}^{(j)}\right) = \frac{1}{h^D}\sum_{i=1}^{N_e} w^{(i)} \mathcal{K}\left(\frac{\|\boldsymbol{\theta}^{(j)} - \boldsymbol{\theta}^{(i)}\|_2}{h}\right), \tag{D.1}$$

where D is the dimensionality of the parameter space, h is the bandwidth, $\mathcal{K}(\cdot)$ is the kernel function (we use the Epanechnikov kernel), and $w^{(i)}$ is the normalized particle weights satisfying $\sum_{i=1}^{N_e} w^{(i)} = 1$.

The choice of bandwidth h is critical, as it governs the trade-off between bias and variance in the density estimate. We use Silverman's rule of thumb to determine h:

$$h = 0.9 \cdot \min\left(\sigma, \frac{\text{IQR}}{1.349}\right) \cdot N_e^{-1/5}, \tag{D.2}$$

where σ is the standard deviation, and IQR is the interquartile range of the particle ensemble.

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press

We leverage Fast Fourier Transform convolution for efficient computation of KDE on a regular grid, which is particularly advantageous given the high dimensionality of our parameter space and the need for rapid entropy evaluation during RL training and online path planning. Despite this efficiency, the computational cost remains significant. To mitigate this, we approximate entropy using only the three most important parameters—emission rate Q and source coordinates $[x_s, y_s]$—while retaining the original particle weights. This simplification strikes a balance between computational feasibility and the need for accurate and informative reward signals.

E. Neural network input

The input to the neural network comprises several components that encode the agent's belief state in a structured format.

Following the approach of Loisy and Eloy (2022), independently developed by Heinonen et al. (2023), and subsequently applied in Loisy and Heinonen (2023) and van Hove et al. (2024), the belief over the source location is represented as a discretized probability distribution centered on the agent. Specifically, the particle ensemble is projected onto a two-dimensional grid of size $(2N_x - 1) \times (2N_y - 1)$, where each grid cell contains the normalized fraction of particles located within that cell. The grid is centered on the agent's current horizontal position $[x, y]$, ensuring translational invariance and enabling the network to generalize across spatial contexts.

The agent's own position and its grid cell visitation history are encoded in a 3D array of size $(2N_x - 1) \times (2N_y - 1) \times (2N_z - 1)$, also centered on the agent's current location $[x, y, z]$. Each cell stores the normalized frequency of visits to that grid cell, providing the network with spatial memory of previously sampled regions.

The belief over wind direction is encoded as a one-hot vector, where the mean wind direction is discretized into bins of $10\,^\circ$.

All components are concatenated and flattened to form the final input vector to the neural network.

F. Synthetic wind observations

Figure F1 visualizes example synthetic wind observations used in this study.

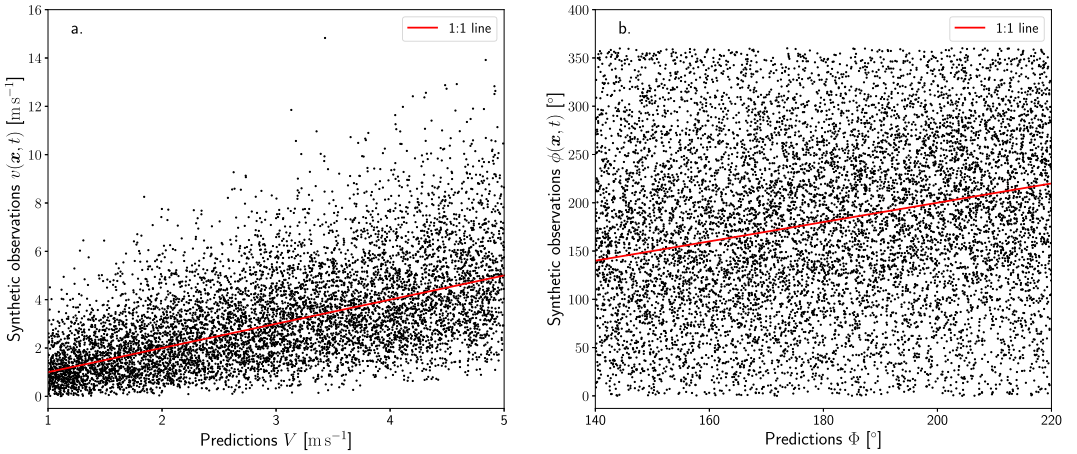


Figure F1. *(a) Example turbulent synthetic wind speed observations $v(\boldsymbol{x}, t)$ obtained via the inverse Box–Cox transformation of $V^{(\lambda)} + \epsilon_V^{(\lambda)}$, with $\lambda = 0.38$ and $\epsilon_V^{(\lambda)} \sim \mathcal{N}\left(0, \sigma_V^{(\lambda)}\right)$, where $\sigma_V^{(\lambda)} = 0.83$. All observations are positive because the Box–Cox transformation is defined exclusively for strictly positive input values, thereby ensuring that transformed wind speed measurements remain > 0. (b) Example turbulent synthetic wind direction observations $\phi(\boldsymbol{x}, t)$ obtained via the inverse Box–Cox transformation of $\Phi^{(\lambda)} + \epsilon_\Phi^{(\lambda)}$, with $\lambda = 1$ and $\epsilon_\Phi^{(\lambda)} \sim \mathcal{N}\left(0, \sigma_\Phi^{(\lambda)}\right)$, where $\sigma_\Phi^{(\lambda)} = 98.09$. Observations exceeding 360° (< 4% of plotted cases) are folded over the $0 - 360\,^\circ$ range by a modulus operation.*

Cite this article: van Hove A, Aalstad K and Pirk N (2026). Actively inferring methane sources with drones. *Environmental Data Science*, 5: e2. doi:10.1017/eds.2026.10029

https://doi.org/10.1017/eds.2026.10029 Published online by Cambridge University Press