

# Multi-Agent Clarity-Aware Dynamic Coverage with Gaussian Processes

Devansh R. Agrawal and Dimitra Panagou

**Abstract**—This paper presents two algorithms for multi-agent dynamic coverage in spatiotemporal environments, where the coverage algorithms are informed by the method of data assimilation. In particular, we show that by explicitly modeling the environment using a Gaussian Process (GP) model, and considering the sensing capabilities and the dynamics of a team of robots, we can design an estimation algorithm and multi-agent coverage controller that explores and estimates the state of the spatiotemporal environment. The uncertainty of the estimate is quantified using clarity, an information-theoretic metric, where higher clarity corresponds to lower uncertainty. By exploiting the relationship between GPs and Stochastic Differential Equations (SDEs) we quantify the increase in clarity of the estimated state at any position due to a measurement taken from any other position. We use this relationship to design two new coverage controllers, both of which scale well with the number of agents exploring the domain, assuming the robots can share the map of the clarity over the spatial domain via communication. We demonstrate the algorithms through a realistic simulation of a team of robots collecting wind data over a region in Austria.

Code and open-sourced Julia packages are available at [1].

## I. INTRODUCTION

A standard robotic mission is the collection of information that varies both in time and space over a domain of interest. To collect such information optimally, a (team of) robot(s) must reason about the currently available information, the target level of confidence in the information sought, the spatiotemporal evolution of the underlying information, and the robot's sensing capabilities, and (in the case of a team) coordinate the actions of each robot.

The design of informative path planners and dynamic coverage controllers has long been of interest [2]–[4], with a variety of techniques proposed including Voronoi partitioning [5], sampling approaches [6], [7], grid/graph based approaches [8], [9] and ergodic search [10], [11].

In this paper, we define the informative path planning or coverage control problem as follows: we have a team of robots that, at a fixed sampling frequency, measure the spatiotemporal environment at their respective positions. Using these measurements, we update our estimate of the state of the environment (referred to as information assimilation), while simultaneously controlling the robots position to determine the next location from which a measurement should be taken (referred to as the coverage controller).

\*The authors would like to acknowledge the support of the National Science Foundation (NSF) under grant no. 1942907 and grant no. 2223845.

Devansh R Agrawal is with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, USA. [devansh@umich.edu](mailto:devansh@umich.edu).

Dimitra Panagou is with the Department of Robotics and the Department of Aerospace Engineering, University of Michigan, Ann Arbor, USA. [dpanagou@umich.edu](mailto:dpanagou@umich.edu).

As such, the goal is to design a controller and information assimilation algorithm that efficiently reduce the uncertainty of the estimate of the state of the environment.

We quantify the uncertainty of a stochastic variable using an information-theoretic metric *clarity*, introduced in [12]. In particular, as the uncertainty of the stochastic variable decreases, (i.e., its differential entropy approaches  $-\infty$ ), the clarity of the random variable approaches 1. Similarly, as the uncertainty increases, the clarity approaches zero.

We model the environment as a spatiotemporal field  $f(t, p)$ , i.e., a scalar function that varies in time and space; as an example, if the goal is to estimate the windspeed over a spatial and temporal domain,  $f(t, p)$  represents the windspeed at any given time  $t$  and position  $p$ . The estimate is a function  $\hat{f}(t, p)$  for each  $t, p$ , with an associated clarity  $q(t, p)$  at each  $t, p$ . Numerically the state of the environment is a vector representing  $\hat{f}(t, p)$  at a set of grid points. By taking (noisy) measurements of  $f$  using the robots at their respective locations, we can improve our estimate  $\hat{f}$  and increase its clarity (i.e., reduce the uncertainty). At the same time, due to the time-varying nature of  $f$ , the clarity of  $\hat{f}$  decreases for all points not being measured. This balance of information gain and decay will be an important element in designing the algorithms.

A key limitation of many of the methods listed above is that simplified heuristics are used to motivate the cost functions used in the informative path planners. For example, the ergodic search approaches assume that a Target Spatial Distribution (TSD) (defined as the desired percentage of time that the robot should spend at any position in the domain) is provided by the user. However, there has been less work on how one can obtain such a target distribution in a principled manner taking into account the sensing capabilities of the robot or the temporal evolution of the state of the environment.

The goal of this paper is to demonstrate how the cost function in informative path planning can be designed in a principled manner based on the assumed model of the environment. In particular, when estimating a spatiotemporal field, a common practice is to model it as a realization of a GP [13], and use the robot's measurements to update the estimate of the state of environment.

Here, we use the connection between GPs and SDEs [14]–[16] to analyze the information-gathering capabilities of the robots: given the robot's take measurements at their respective locations, how much does the uncertainty in our estimate of state of the environment reduce? We answer this by quantifying a robot's sensing function and the environment's information decay function. For a point  $p$  in the domain,

the sensing function defines the rate of increase of clarity at  $p$  due to measurements from a robot at position  $r$ . The decay function quantifies the rate of decrease of clarity due to the time-varying nature of  $f(t, p)$ . We use these functions to design coverage controllers that respect the rate of change of clarity when designing trajectories.

This paper has three main contributions: (A) We use clarity [12] to quantify the rate of change of uncertainty at a position  $p$  due to measurements made by a robot at a (possibly different) position  $r$ . Integrated over the mission domain, this quantifies the value of the robot being at position  $r$ . (B) We use this relation to propose two coverage controllers. (C) Being feedback controllers, we show how they scale naturally to the multi-agent setting. Finally, we demonstrate the algorithms using a realistic simulation, where a team of aerial robots explore a region of Austria, and estimate the wind speed over this region.

The two coverage algorithms proposed bear resemblance to the controllers in [4] and [17]. The first, referred to as the *direct controller*, directly chooses a control input to maximize the clarity of the state of the environment. The second, referred to as the *indirect controller*, computes a TSD based on the time required to increase the clarity to a given target value.

## II. PRELIMINARIES

**Notation:**  $\mathbb{Z}$  is the set of integers,  $\mathbb{N} = \{0, 1, 2, \dots\}$  is the set of naturals.  $\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{> 0}$  denote the sets of reals, nonnegative reals, and positive reals.  $I_N$  denotes the  $N$ -dimensional identity matrix.  $\mathbb{S}_{++}^n$  denotes the set of symmetric positive definite matrices in  $\mathbb{R}^{n \times n}$ . For  $A \in \mathbb{S}_{++}^n$ ,  $\sqrt{A} \in \mathbb{S}_{++}^n$  is the unique matrix such that  $\sqrt{A}\sqrt{A} = A$ . For  $v \in \mathbb{R}^n$ , the  $i$ -th entry is  $[v]_i$ . Similarly, for  $M \in \mathbb{R}^{N \times M}$ , the  $(i, j)$ -th entry is  $[M]_{(i, j)}$ .  $A \otimes B$  denotes the Kronecker product of  $A, B$ .

We consider a problem with  $N_R$  robots, exploring a  $d$ -dimensional domain  $\mathcal{D} \subset \mathbb{R}^d$ . Each robot has a state in  $\mathcal{X} \subset \mathbb{R}^n$ ,  $n \geq d$ . The state of the environment will be represented numerically at a set of  $N_G$  grid points.

### A. Clarity

The information metric *clarity* was introduced in [12] and is based on differential entropy:

**Definition 1.** [18, Ch. 8] The *differential entropy*  $h[X] \in (-\infty, \infty)$  of a continuous random variable  $X$  with support  $S$  and density  $\rho : S \rightarrow \mathbb{R}$  is

$$h[X] = - \int_S \rho(x) \log \rho(x) dx. \quad (1)$$

Notice that as the uncertainty in  $X$  decreases, the entropy approaches  $h[X] \rightarrow -\infty$ . Clarity is defined in terms of differential entropy.

**Definition 2.** Let  $X$  be a  $n$ -dimensional continuous random variable with differential entropy  $h[X]$ . The *clarity*  $q[X] \in (0, 1)$  of  $X$  is defined as:

$$q[X] = \left( 1 + \frac{\exp(2h[X])}{(2\pi e)^n} \right)^{-1}. \quad (2)$$

In other words, the clarity  $q[X]$  about a random variable  $X$  lies in  $(0, 1)$ , where  $q \rightarrow 0$  corresponds to the case where the uncertainty in  $X$  is infinite, and if  $X$  is perfectly known in an idealized (noise-free) setting,  $q[X] = 1$ . For a scalar Gaussian random variable  $X \sim \mathcal{N}(\mu, \sigma^2)$ , the clarity is  $q[X] = 1/(1 + \sigma^2)$ .

In an estimation context, we use clarity to quantify the quality of our estimate: as the clarity increases towards 1, the uncertainty of our estimate decreases towards 0. In [12] it was shown that when  $X$  is estimated using a Kalman filter, the clarity dynamics of the estimate of  $X$  can be obtained in closed form.

### B. Gaussian Processes

A GP [13, Ch. 2] is a (scalar) stochastic process that is fully defined by the mean function  $\mathbf{m} : \mathcal{D} \rightarrow \mathbb{R}$  and a kernel  $\mathbf{k} : \mathcal{D} \times \mathcal{D} \rightarrow \mathbb{R}$ :

$$f(p) \sim \text{GP}(\mathbf{m}(p), \mathbf{k}(p, p')), \quad (3)$$

where  $\mathbf{m}$  and  $\mathbf{k}$  are defined as

$$\mathbf{m}(p) = E[f(p)], \quad (4a)$$

$$\mathbf{k}(p, p') = E[(f(p) - \mathbf{m}(p))(f(p') - \mathbf{m}(p'))]. \quad (4b)$$

Given a set of  $N$  measurements  $\{y_k\}_{k=1}^N$  taken at positions  $\{p_k\}_{k=1}^N$ , we can update our posterior estimate of  $f$ , as described in [13, Ch. 2].

For two set of points  $P_A = \{a_i\}_{i=1}^N$  and  $P_B = \{b_i\}_{i=1}^M$ , the kernel matrix  $\mathbf{K}_{AB} \in \mathbb{R}^{N \times M}$  is the matrix such that  $[\mathbf{K}_{AB}]_{(i, j)} = \mathbf{k}(a_i, b_j)$ .

### C. Spatiotemporal Gaussian Processes

The goal is to estimate a spatiotemporal field, i.e., to estimate a function  $f(t, p)$ ,  $f : \mathbb{R} \times \mathcal{D} \rightarrow \mathbb{R}$  using measurements obtained by robots.<sup>1</sup> Here  $t \in \mathbb{R}$  denotes time, and  $\mathcal{D} \subset \mathbb{R}^d$  is spatial domain of interest. The measurements (defined in (13)) are noisy measurements of  $f$  at a fixed sampling period from each robot's position at the sampling time.

While a standard GP can directly handle the spatiotemporal case, we can achieve significant computational efficiency by explicitly separating the spatial and temporal dimensions and exploiting the equivalence between spatiotemporal GPs and SDEs. Effectively, we can convert a Bayesian inference problem into a Kalman Filtering problem, thereby reducing memory and computational cost. We assume the following:

**Assumption 1.** Suppose the spatiotemporal field  $f : \mathbb{R} \times \mathcal{D} \rightarrow \mathbb{R}$  is a realization of a zero-mean GP:

$$f(t, p) \sim \text{GP}(0, \mathbf{k}(t, p, t', p')), \quad (5)$$

$$\mathbf{k}(t, p, t', p') = \mathbf{k}_T(t, t') \mathbf{k}_S(p, p'), \quad (6)$$

where the kernel is separable in space and time, and the temporal kernel is isotropic, i.e.,  $\mathbf{k}_T(t, t')$  only depends on  $|t' - t|$ .

<sup>1</sup>For simplicity of exposition, we assume the spatiotemporal field has scalar outputs. For multidimensional outputs, we repeat for each dimension independently.

Under Assumption 1, it is known that realizations of a GP are also realizations of a SDE [16]. This fact is derived through the Wiener-Khinchin theorem [19, Ch. 12], and in the interest of space, the readers are referred to [16] or [20, Appendix] for full derivations.

The key idea is that if  $h(t) \sim \text{GP}(0, k_T(t, t'))$  is a realization of a (temporal) GP, it is equal to the output of a transfer function applied to a realization of a white noise process. By expressing the transfer function in state-space form, we arrive at a SDE such that a realization of the SDE is equal to  $h$ .

In the spatiotemporal case, let  $P_G = \{p_i\}_{i=1}^{N_G} \subset \mathcal{D}$  be a set of  $N_G$  (possibly non-uniform) grid points over the spatial domain. Let  $\mathbf{f}(t) \in \mathbb{R}^{N_G}$  be a vector such that the  $i$ -th entry is the value of the spatiotemporal field at the  $i$ -th grid point,  $[\mathbf{f}(t)]_i = f(t, p_i)$ . Then, the SDE for the system comprises of  $N_G$  independent stochastic processes (7a), that get spatially correlated based on the spatial kernel (7b). Mathematically,

$$\begin{cases} ds_i(t) = As_i(t)dt + BdW_i(t) \\ z_i(t) = Cs_i(t) \\ s_i(0) \sim \mathcal{N}(0, \Sigma) \end{cases}, \quad (7a)$$

$$\mathbf{f}(t) = \sqrt{K_{GG}}\mathbf{z}(t) = \sqrt{K_{GG}}(I_{N_G} \otimes C)\mathbf{s}, \quad (7b)$$

Here  $s_i(t) \in \mathbb{R}^{n_k}$  is a state at each grid point.<sup>2</sup>  $\mathbf{s} = [s_1^T \ \dots \ s_G^T]^T \in \mathbb{R}^{n_k N_G}$  is a stacked vector representing the state of the entire environment;  $\mathbf{f}(t) = [f(t, p_1) \ \dots \ f(t, p_G)]^T \in \mathbb{R}^{N_G}$  is a stacked vector comprising the value of the field at each grid point;  $W_i$  is a standard Wiener process, independent for each grid point. The matrices  $A \in \mathbb{R}^{n_k \times n_k}$ ,  $B \in \mathbb{R}^{n_k \times 1}$ ,  $C \in \mathbb{R}^{n_k \times l}$  are constant matrices that only depend on the temporal kernel  $k_T$ .  $\Sigma \in \mathbb{S}_{++}^{n_k}$  is the matrix that solves  $A\Sigma + \Sigma A^T = -BB^T$ .  $K_{GG} \in \mathbb{S}_{++}^G$  is the spatial kernel matrix, i.e.,  $[K_{GG}]_{ij} = k_S(p_i, p_j)$ .

*Example 1.* The Matern-1/2 temporal kernel is  $k_T(t, t') = \sigma_t^2 \exp(-\lambda_t |t - t'|)$  for hyperparameters  $\lambda_t, \sigma_t > 0$ . The state-space model has dimension  $n_k = 1$ , and matrices  $A = [-\lambda_t]$ ,  $B = [1]$ ,  $C = [\sqrt{2\lambda_t}\sigma_t]$ . Derivations and expressions for Matern-3/2 and Matern-5/2 kernels can be found in [20, Appendix].

#### D. Ergodic Control

Ergodic control [10], [11] is a technique to generate robot trajectories that cover a domain  $\mathcal{D} = [0, L_1] \times \dots \times [0, L_d] \subset \mathbb{R}^d$ , such that the trajectories have a spatial (position) distribution that closely matches a specified TSD, as explained below.

The TSD is a function  $\phi : \mathcal{D} \rightarrow \mathbb{R}$  such that the  $\phi(p)$  denotes the desired time a robot should spend at position  $p$ . Given a robot's (position) trajectory  $\xi : [0, T] \rightarrow \mathcal{D}$  the trajectory's spatial distribution is defined as  $c_\xi : \mathcal{D} \rightarrow \mathbb{R}$ ,

<sup>2</sup> $n_k$  depends on the temporal kernel. For the Matern 1/2, 3/2, and 5/2 kernels,  $n_k = 1, 2, 3$  respectively.

where for any  $p \in \mathcal{D}$

$$c_\xi(p) = \frac{1}{T} \int_0^T \delta(p - \xi(\tau)) d\tau. \quad (8)$$

Here  $\delta : \mathbb{R}^d \rightarrow \mathbb{R}$  is the Dirac delta function.

The *ergodicity*  $E > 0$  of a trajectory  $\xi$  measures the difference between the robot trajectory's spatial distribution and the target spatial distribution:

$$E = \|c_\xi - \phi\|_{H^{-s}}^2 \quad (9)$$

where  $\|\cdot\|_{H^{-s}}$  is the Sobolev space norm of order  $s = (d+1)/2$ , defined in [10]:

$$\|c_\xi - \phi\|_{H^{-s}}^2 = \sum_{l \in \mathbb{N}^d} \Lambda_l (\hat{c}_l - \hat{\phi}_l)^2 \quad (10)$$

where  $\Lambda_l \in \mathbb{R}$  is a weighting coefficient, and  $(\cdot)_l$  is the  $l$ -th element of the Discrete Cosine Transform (DCT) of the function  $(\cdot)$ , e.g.

$$\hat{\phi}_l = \langle \phi_l, b_l \rangle = \int_{p \in \mathcal{D}} \phi_l(p) b_l(p) dp \quad (11)$$

where  $b_l : \mathcal{D} \rightarrow \mathbb{R}$  is the  $l$ -th basis function. We refer the reader to [10] for further details.

$E$  is a function-space norm measuring the difference between the TSD and the spatial distribution of the trajectory. The key benefit of the Sobolev norm is that it prioritizes matching the low spatial frequency differences between  $c$  and  $\phi$  before matching the high spatial frequencies. This means that the controllers have a multiscale-spectral nature, where they prioritize covering the domain globally, before returning to the gaps and covering them [10].

In [10] a feedback controller is derived for single and double-integrator robot models that minimizes the ergodicity. Various extensions have been presented in, for example, [11], [21] to address other robot models and other goals.

### III. PROBLEM STATEMENT

Consider a team of  $N_R > 0$  robots, each with dynamics

$$\dot{x}_i = F(x_i) + G(x_i)u_i, \quad (12)$$

where  $x_i \in \mathcal{X} \subset \mathbb{R}^n$  is the  $i$ -th robot's state, and  $u_i \in \mathcal{U} \subset \mathbb{R}^m$  is its control input. The position of each robot is  $r_i = \Phi(x_i) \in \mathbb{R}^d$ , i.e.,  $\Phi : \mathcal{X} \rightarrow \mathcal{D}$  extracts the position.

Each robot makes measurements of the spatiotemporal field at a fixed sampling period  $\Delta T > 0$ ,

$$y_{k,i} = f(t_k, \Phi(x_i(t_k))) + w_{k,i}, \quad (13a)$$

$$w_{k,i} \sim \mathcal{N}(0, \sigma_m^2), \quad (13b)$$

that is,  $y_{k,i} \in \mathbb{R}$  is a scalar measurement output by the  $i$ -th robot at the  $k$ -th timestep,  $t_k = k\Delta T$ . Each measurement is perturbed by zero-mean Gaussian noise with standard deviation  $\sigma_m$ .

We assume each robot determines its control inputs, but that the information from each robot is assimilated centrally. We assume the robots can always communicate with the

central agent, sending the measurements and receiving a map of the current clarity at each  $p \in \mathcal{D}$ .

**Problem 1.** Consider a team of  $N_R > 0$  robots, each with dynamics (12) and measurements (13), exploring a domain  $\mathcal{D}$ . Let  $f : \mathbb{R} \times \mathcal{D} \rightarrow \mathbb{R}$  be a spatiotemporal field to be estimated satisfying Assumption 1. Design a coverage control algorithm for each robot, and an estimation algorithm to fuse measurements  $y_k$  into an estimate of  $f$ .

The mathematical form of the coverage objective is delayed until Section V. The estimator will be the optimal estimator in a least-squares sense, discussed in Section IV.

In addressing Problem 1, we address two questions: (A) how does the information assimilation algorithm inform the value of taking measurements at a robot position  $x \in \mathcal{D}$  on the quality of information at a different position  $p \in \mathcal{D}$ , and (B) how should one design coverage controllers to exploit that relationship? Since the mission is a multi-agent coverage problem, we also need to ensure that the proposed coverage algorithms are scalable with the number of robots. We address these two questions in the following sections.

#### IV. INFORMATION ASSIMILATION

In this section, we discuss how the GP model (Assumption 1) determines two functions: (A) the information decay rate at each  $p \in \mathcal{D}$ , and (B) the information gain rate at each  $p \in \mathcal{D}$  due to measurements taken from a robot's position  $r_i = \Phi(x_i) \in \mathcal{D}$ . We consider the hyperparameters of the GP to be specified and constant, although some strategies for estimating these are discussed in the simulation section.

##### A. Kalman Filter Model

First, we show that the Kalman Filter (KF) is the optimal state estimator to estimate the spatiotemporal field  $f$ . As shown in Section II-C, the process model for  $f$  sampled at  $N_G$  grid points is a linear stochastic differential equation with state  $s \in \mathbb{R}^{n_k N_G}$ . We now show that the measurements (13) are a linear function of  $s$ .<sup>3</sup>

Consider  $N_R$  robots at positions  $P_R = \{\Phi(x_i)\}_{i=1}^{N_r}$  where each robot makes a measurements  $y_{k,i}$  as in (13). However, the state  $s$  corresponds to the grid points  $P_G$ , not necessarily coinciding with the measurement locations  $P_R$ . To account for this, we use spatial correlation based on the Gaussian Process model for  $f$ :

$$\begin{bmatrix} f(t_k) \\ y_k \end{bmatrix} \sim \mathcal{N} \left( 0, \begin{bmatrix} K_{GG} & K_{GR} \\ K_{RG} & K_{RR} + \sigma_m^2 I \end{bmatrix} \right) \quad (14)$$

where  $K_{GG}, K_{GR}, K_{RG}, K_{RR}$  are the kernel matrices for the sets of points  $P_G, P_R$ , and  $y_k = [y_{k,1} \ \dots \ y_{k,N_R}]^T$ .

Using (7b),  $y_k$  conditioned on the state  $s(t_k)$  is

$$y_k | s(t_k) \sim \mathcal{N}(Hs(t_k), V), \quad (15a)$$

$$H = K_{RG} K_{GG}^{-1} \sqrt{K_{GG}} (I_{N_G} \otimes C) \quad (15b)$$

$$V = \sigma_m^2 I_{N_R} + K_{RR} - K_{RG} K_{GG}^{-1} K_{GR}. \quad (15c)$$

<sup>3</sup>In [16], the measurements must be taken at one of the grid points. Here we extend the result to allow measurements at non-grid points.

Therefore, the environment's state space model is a linear (continuous time) process (recall (7a)) with linear (discrete-time) measurements:

$$ds = (I_{N_G} \otimes A)sdt + (I_{N_G} \otimes B)dW, \quad (16a)$$

$$y_k = Hs(t_k) + v_k \quad (16b)$$

where  $W$  is a  $N_G$ -dimensional standard Wiener process, and  $v_k \sim \mathcal{N}(0, V)$ . Notice that although each measurement has noise variance  $\sigma_m^2 I$ , the noise model in (16b) has  $V \geq \sigma_m^2 I$  accounting for the fact that measurements can be taken at non-grid points.

To summarize, we have a linear, time-invariance process model (16a), with a linear (but time-varying due to the changing measurement locations) measurement model (16b). Together, they satisfy the assumptions of the KF, and therefore the KF is the optimal estimator for this system [22].

##### B. Quantifying Information Gain and Decay Rates

Next, we wish to characterize the clarity dynamics, i.e., the rate of information gain and decay. In this section, we focus on the clarity dynamics of a single point  $p \in \mathcal{D}$  due to a measurement taken by a robot with position  $r = \Phi(x) \in \mathcal{D}$ . Since we use the KF to assimilate measurements, we use the earlier derived dynamics to estimate the rate of information gain. Reducing (16) for a single point  $p$ , the continuous time KF model is

$$\dot{s} = As + Bw, \quad w(t) \sim \mathcal{N}(0, I), \quad (17a)$$

$$y = Ls + v, \quad v(t) \sim \mathcal{N}(0, V\Delta T) \quad (17b)$$

where  $s \in \mathbb{R}^{n_k}$  is the state of the spatiotemporal process at  $p$ ,  $r = \Phi(x)$  is the robot's position, and

$$L = \frac{k_s(r, p)}{\sqrt{k_s(p, p)}} C, \quad V = \sigma_m^2 + k_s(r, r) - \frac{k_s(r, p)^2}{k_s(p, p)}.$$

Let the KF state consist of  $(\hat{s}, \Sigma)$ , the mean and covariance. Then, the covariance has dynamics

$$\dot{\Sigma} = A\Sigma + \Sigma A^T + BB^T - \Sigma L^T (V\Delta T)^{-1} L\Sigma. \quad (18)$$

Therefore, the estimate of  $f(t, p)$  is  $\mathcal{N}(\hat{f}, \Pi)$ , where  $\hat{f} = C\hat{s}$ , and  $\Pi = C\Sigma C^T$ . Since, the clarity of a scalar Gaussian variable is  $q = 1/(1 + \Pi)$ , the clarity dynamics are

$$\dot{q} = \frac{dq}{d\Pi} \dot{\Pi} = -q^2 C \dot{\Sigma} C^T. \quad (19)$$

Depending on the temporal kernel,<sup>4</sup> this simplifies to

$$\dot{q} = \underbrace{S(x, p)(1 - q)^2}_{\text{clarity gain}} - \underbrace{D(p, q)}_{\text{clarity decay}} \quad (20)$$

where the first term defines the rate of clarity gain at  $p$  due to measurements taken at  $r = \Phi(x)$ , while the second term defines the clarity decay rate.

**Remark 1.** Eq. (20) is one of our main results: the function  $S : \mathcal{X} \times \mathcal{D} \rightarrow \mathbb{R}$  is the sensing function that quantifies the importance of a measurement taken from robot state  $x \in \mathcal{X}$

<sup>4</sup>In particular, this holds for Matern-1/2 kernels.

on the clarity of our estimate at a position  $p \in \mathcal{D}$ . Similarly,  $D : \mathcal{D} \times \mathbb{R} \rightarrow \mathbb{R}$  defines the rate at which clarity about  $f(t, p)$  decays due to the spatiotemporal nature of  $f$ . Notice the decay rate is uncontrolled, i.e., does not depend on the robot's state  $x$ .

*Example 2.* For Matern-1/2 temporal kernels,

$$S(x, p) = \frac{1}{\Delta T} \frac{k_S(r, p)^2}{k_S(p, p)(k_S(r, r) + \sigma_m^2) - k_S(r, p)^2}$$

$$W(p, q) = 2\lambda_t ((\sigma_t^2 + 1)q^2 - q),$$

where  $r = \Phi(x)$  is the position of a robot at state  $x$ . Since for isotropic spatial kernels  $k_S(p, p') = k_S(\|p - p'\|)$ ,

$$S(d) \propto \frac{k_S(d)^2}{k_S(0)^2 + \sigma_m^2 k_S(0) - k_S(d)^2}$$

where  $d = \|\Phi(x) - p\|$  is the distance at which the measurement is taken. When  $d \mapsto k_S(d)$  is nonincreasing, e.g. in the Matern and Squared Exponential kernels,  $S(x, p)$  is maximized at  $\Phi(x) = p$ , implying that the rate of increase in clarity about  $p$  is maximized when the robot is also at position  $p$ . This is not, in general, true, since for example in periodic or polynomial spatial kernels,  $S(x, p)$  may be maximized for some  $\Phi(x) \neq p$ . Furthermore notice that in the limiting case of a spatiostatic environment,  $\lambda_t \rightarrow 0$ , and therefore the decay rate  $D(p, q) \rightarrow 0$ .

To summarize, in this information-gathering problem the spatiotemporal information to be collected is modeled using a GP. To define a suitable coverage algorithm, we need to quantify the value of taking a measurement at some robot state  $x \in \mathcal{X}$  on the clarity gain at any other position  $p \in \mathcal{D}$ . This is captured by the clarity dynamics (20). The key functions are  $S$ , the sensing function, and  $W$ , the decay function. Notice only  $S(x, p)$  is controllable since it is the only term in (20) that depends on the robot's state  $x$ .

## V. COVERAGE CONTROLLERS

In this section, we use the sensitivity and decay functions in (20) to derive two coverage controllers. The direct method chooses a control input that maximizes the rate of increase in the total clarity integrated over the domain  $\mathcal{D}$ . The indirect method determines the time that the robot should spend at each position in the domain to achieve a target clarity and then uses ergodic control to compute the control input.

### A. Direct Method

The direct method minimizes the cost function

$$J(t) = \|\bar{q}(\cdot) - q(t, \cdot)\|_H^2, \quad (21)$$

a function-space norm over  $p \in \mathcal{D}$  between the current clarity distribution  $q(t, p)$  and the target clarity distribution  $\bar{q}(p)$ . We use the Sobolev norm, defined in (10), and discussed below.

Notice that  $J(t)$  does not explicitly depend on the robot's state or control input. As such, we choose to minimize  $J$  over a short horizon  $\delta > 0$  in the future:

$$J(t + \delta) \approx J(t) + \dot{J}(t, x)\delta^2 + \frac{1}{2}\ddot{J}(t, x, u)\delta^2 + \dots \quad (22)$$

where the dependency on  $u$  first shows up in the  $\ddot{J}$  term. This high-relative degree behavior is a consequence of the fact that the clarity dynamics (20) depend on  $x$ , not  $\dot{x}$ . Therefore the second derivative of  $J$  must be taken for the control input to appear in the expressions. This behavior is commonly observed in the literature on coverage control, as in [4, Ch. 2] and in [10]. Then, given control inputs  $u \in \mathcal{U} \subset \mathbb{R}^m$ , the controller will be of the form

$$\pi(t, x) = \operatorname{argmin}_{u \in \mathcal{U}} \ddot{J}(t, x, u). \quad (23)$$

We will derive a closed-form solution for this controller. Before doing so, we justify our choices for the cost function and the control strategy.

We use the Sobolev norm for the following reasons. In [4], a differentiable sensing functional (an analog of  $S$ ) is used with the generalized transport theorem to compute an analog of  $\ddot{J}(t, x, u)$ . However, this approach often leads to local minima, where  $\ddot{J}(t, x, u)$  becomes independent of  $u$ . This happens when all of the local information has been collected, and there is no preference for the controller to move in one direction over the other. To address this, [4] proposed combining the local search strategy with a global strategy, where the controller would choose a new global waypoint when the local controller reaches a local minimum. In our work, we use the Sobolev space norm instead of the  $\ell_2$  norm, and this allows the controllers to have a multispectral property [10] - it prioritizes global coverage before prioritizing local coverage.

Second, to evaluate  $\ddot{J}$ , we use the clarity dynamics we derived in Section IV-B. This is in contrast to earlier works that used heuristic expressions to quantify coverage, and coverage dynamics [4], [23]. As such, the derived controllers depend explicitly on the spatiotemporal field's kernel, and the sensing capabilities (in particular the sampling period  $\Delta T$  and measurement noise  $\sigma_m$ ) of the robots.

Next, we derive the controller. The cost function is

$$J(t) = \|\bar{q}(\cdot) - q(t, \cdot)\|_H^2 = \sum_{l \in \mathbb{N}^d} \Lambda_l (\hat{q}_l - \hat{q}_l(t))^2, \quad (24)$$

where  $\hat{q}_l = \langle \bar{q}, b_l \rangle$ ,  $\hat{q}_l(t) = \langle q(t, \cdot), b_l \rangle$  are the inner products of  $\bar{q}(\cdot)$  and  $q(t, \cdot)$  with the  $l$ -th basis function of the DCT. Recall the notation  $\langle a, b_l \rangle$ , and  $\Lambda_l$  was defined in Section II-D. After some algebraic calculations, one can show that the first and second time-derivatives of  $J$  are:

$$\dot{J}(t, x) = \sum_{l \in \mathbb{N}^d} -2\Lambda_l (\hat{q}_l - \hat{q}_l(t)) \dot{\hat{q}}_l(t, x)$$

$$\ddot{J}(t, x, u) = \sum_{l \in \mathbb{N}^d} 2\Lambda_l \left( \ddot{\hat{q}}_l^2(t, x) - (\hat{q}_l - \hat{q}_l(t)) \ddot{\hat{q}}_l(t, x, u) \right)$$

where  $\dot{\hat{q}}_l(t, x)$ ,  $\ddot{\hat{q}}_l(t, x, u)$  are

$$\begin{aligned} \dot{\hat{q}}_l &= \frac{d}{dt} \langle q(t, \cdot), b_l \rangle \\ &= \int_{p \in \mathcal{D}} (S(x, p)(1 - q(t, p))^2 - W(p, q(t, p))) b_l(p) dp \end{aligned}$$

where  $S$  is as defined in (20). Similarly,

$$\ddot{\hat{q}}_l = \frac{d^2}{dt^2} \int_{p \in \mathcal{D}} q(t, p) b_l(p) dp = \hat{B}_l(t, x) \dot{x} + \mathcal{O},$$

where  $\mathcal{O}$  collects terms independent of  $\dot{x}$  (and therefore  $u$ ), and  $\hat{B}_l(t, x) \in \mathbb{R}^{1 \times n}$  is as defined as

$$\hat{B}_l(t, x) = \left\langle (1 - q(t, \cdot))^2 \frac{\partial S}{\partial x}(x, \cdot), b_l \right\rangle. \quad (25)$$

Therefore, we have

$$\begin{aligned} \ddot{J}(t, x, u) &= \sum_{l \in \mathbb{N}^d} -\Lambda_l(\hat{q}_l - \hat{q}_l(t)) \hat{B}_l(t, x) \dot{x} + \mathcal{O} \\ &= -L(t, x)(F(x) + G(x)u) + \mathcal{O} \end{aligned}$$

where we define

$$L(t, x) = \sum_{l \in \mathbb{N}^d} \Lambda_l(\hat{q}_l - \hat{q}_l(t)) \hat{B}_l(t, x).$$

Therefore, the choice of  $u$  that minimizes  $J(t + \delta)$  yields a feedback controller  $\pi_{dir} : \mathbb{R} \times \mathcal{D} \rightarrow \mathcal{U}$ ,

$$\pi_{dir}(t, x) = \underset{u \in \mathcal{U}}{\operatorname{argmin}} -L(t, x)G(x)u$$

If  $\mathcal{U} = \{u \in \mathbb{R}^m : \|u\| \leq u_{max}\}$ , and  $L(t, x)G(x) \neq 0$ ,

$$\pi_{dir}(t, x) = u_{max} \frac{G(x)^T L^T(t, x)}{\|L(t, x)G(x)\|}. \quad (26)$$

Proving that  $L(t, x)G(x) \neq 0$  for any  $t, x$  is non-trivial, and will be studied in future work.

### B. Indirect Method

The second approach is inspired by ergodic control. Ergodic control uses a TSD to determine the feedback control law, as discussed in Section II-D. Here we derive a principled method to construct the TSD based on the information assimilation algorithm discussed in Section IV.

The key idea is to set the TSD to be the time required for the clarity of our estimate of  $f$  to increase from its current value to a specified target clarity  $\bar{q}(p)$ , assuming the robot was making measurements from  $x = p$ . To compute this, we solve the differential equation (20) and determine  $T(q, \bar{q})$ , i.e., the time required to increase the clarity from  $q$  to  $\bar{q}$ .

Then, given the target clarity distribution  $\bar{q} : \mathcal{D} \rightarrow [0, 1]$ , and the current clarity distribution  $q(t, \cdot) : \mathcal{D} \rightarrow [0, 1]$ , the TSD can be specified as follows:

$$\text{TSD}(t, p) = \begin{cases} T(q(t, p), \bar{q}(p)) & \text{if } q(t, p) \leq \bar{q}(p) \\ 0 & \text{else} \end{cases}. \quad (27)$$

This equation has an analytic solution, see [20, Appendix].

Finally, we can use the ergodic control method described in [10] to design a feedback controller for the system,

$$\pi_{ind}(t, x) = \pi_{ergo}(t, x, \text{TSD}) \quad (28)$$

### C. Extension to Multi-Robot Coverage Control

Our proposed coverage controllers have been presented for the single-robot cases above. Here we discuss the extension

and implementation of these methods in the multi-agent case, where multiple robots have to decide how to move to collect information. We assume that they can synchronize their information by sharing the clarity map,  $q(t, p) \forall p \in \mathcal{D}$ , over a centralized setting, i.e., that they are connected over a complete graph so that each robot has access to a centrally stored clarity map. The extensions to distributed settings are left for future work.

Notice that both proposed controllers are feedback controllers, depending on the robot's position, and the clarity map  $q(t, p)$ . Therefore, the control input for each agent can be computed as  $u_i = \pi(t, x_i, q)$ , where  $x_i$  denotes the position of the  $i$ -th agent, and  $\pi \in \{\pi_{dir}, \pi_{ind}\}$  can be either control strategy. In the indirect approach, we must also share the history of positions visited by the agents.

As the robots move using the coverage controllers, the robots make measurements of the spatiotemporal field from their respective positions. These measurements are assimilated into a single estimate of the spatiotemporal field using the KF model. The information assimilation is currently performed centrally, although future work will look into distributed methods of maintaining the estimate.

## VI. SIMULATIONS

In this section, we report the simulation results of an information-gathering mission. As a prototypical example, we consider the collection of wind data using a team of ten aerial robots. The robots perform a two-hour mission, and we aim to maximize the clarity of the wind field over the domain by the end of the mission. Our evaluation metric is both the accuracy of the reconstruction, as well as the average clarity over the mission domain.

The mission domain is a  $12.7 \times 6.3$  km<sup>2</sup> region of south-eastern Austria, located near  $46.93^\circ$  N,  $15.90^\circ$  E, chosen because of a high-quality ground-truth data set available from WegenerNet [24]. The dataset provides wind speeds over the domain at a resolution of 100 m and 30 minutes. The mission domain is particularly challenging due to its high weather and climate variability [24]. Over the domain considered, the maximum wind speed is 13 m/s.

Each robot is capable of measuring the local x- and y-wind speed every 5 seconds. Each measurement is perturbed by noise with  $\sigma_m = 0.5$  m/s. The robots are modeled as single-integrators with a maximum speed of 15 m/s. We use the KF model with a spatial grid resolution of 320 and 160 m in the x- and y-directions to model the state of the environment.

The spatial and temporal hyperparameters were estimated using techniques from geostatistics [25], [26]. In particular, we constructed a variogram of the dataset and used a least-squares fit to both the Matern-1/2 and the Squared Exponential kernels. The Matern-1/2 kernel fits the data better and is depicted in Figure 1b. The resulting kernel is of the form  $k(t, p, t', p') = \sigma^2 \exp(-|t - t'|/l_t) \exp(-\|x - x'\|/l_s)$ , where  $\sigma = 2.11$  m/s,  $l_t = 183$  min,  $l_s = 1.61$  km. Fitting the kernel using the variogram was computationally much faster and more reliable than the nonlinear minimization of

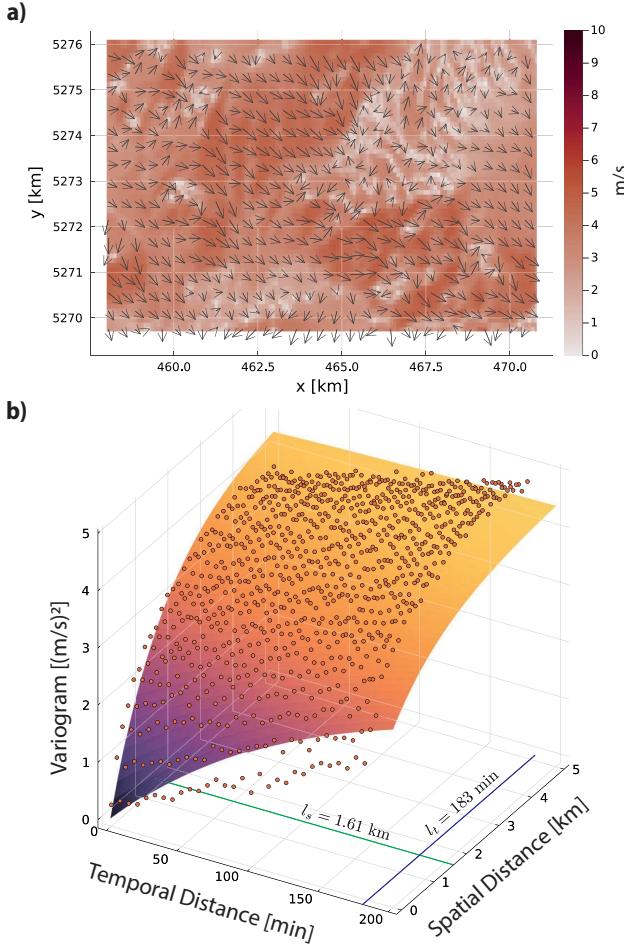


Fig. 1. Wind data from WegenerNet [24]. (a) Wind speed and direction on Jan 1, 2023, 00:00, (b) Variogram showing the spatiotemporal correlation of the data. Surface shows the fitted kernel.

the log-likelihood method of [13]. See [20, Appendix] for additional details.

Simulations were run using both the direct and the indirect control strategies, and the results are summarized in Figure 2. Fig. 2(a) shows the ground-truth data to be estimated.<sup>5</sup>

Fig. 2(b) shows the change in average clarity over time as the robots explore the environment. Both the direct and indirect methods result in an almost identical average clarity at each timestep. Furthermore, after about an hour of exploration, the average clarity reaches a steady state value. This shows that due to the information decay rate, even as the robots continually explore the environment, the average clarity cannot be increased further.

Fig. 2(c,d,f,g) show the trajectories using both controllers. Figs. 2(c, d) show the trajectories of the direct method after 8 and 60 mins, and Figs. 2(f, g) show the corresponding trajectories of the indirect method. The trajectories generated by the two methods are remarkably different - in the direct method, the trajectories are jagged and tend to follow straight lines. This is because of  $\partial S / \partial x$  in (25), which places

<sup>5</sup>In the interest of space, only the x-component of the speed is shown. Refer to [1] for additional figures.

significant benefit on local data collection. In contrast, the indirect method creates smoother trajectories.

Fig. 2(e, h) show the estimated wind speed at  $t=60$  min. Comparing these to the ground truth in Fig. 2(a), is it clear that both methods estimate the wind field accurately.

In Fig. 2(b), we also compare the behavior when using three robots to that of using ten robots. As expected, when there are ten agents the mean clarity is higher (and increases faster) than when there are only three agents.

## VII. CONCLUSIONS

In conclusion, this paper addresses the design of cooperative multi-agent coverage controllers, where the information is shared centrally, but the control decisions are made by each robot independently. We identified a gap between information assimilation algorithms and coverage controllers. Therefore we proposed a method to quantify the value/impact that taking measurements in a domain has on the clarity of our estimate of other parts of the domain. To this end, we utilized Gaussian Processes to model the environment, as well as our earlier work on the clarity dynamics, which in effect quantifies the information gained about the domain due to measurements. We saw that the relative value of measurements is captured by a function  $S$ . We used this function to propose two new coverage controllers that, although qualitatively different, still cover the domain and collect information accurately. The concepts were demonstrated through a simulation study of collecting information about a wind field.

A key limitation of this work is that we assumed the spatial and temporal hyperparameters of the Gaussian Process were fixed and known a priori. Although a method was described to obtain these hyperparameters from data, in our future work we will aim to develop an online method to estimate the hyperparameters and choose trajectories that improve the quality of the hyperparameters. Finally, it would also be interesting to look into methods to ensure the safety of the robots with a safety constraint that depends on the information collected online. In such a scenario, the objective of collecting information must be weighed against the importance of not violating safety constraints.

## REFERENCES

- [1] D. R. Agrawal, "Multiagent clarity-based dynamic coverage," <https://github.com/dev10110/multiagent-clarity-based-dynamic-coverage>, 2024.
- [2] J. Liang, M. Liu, and X. Kui, "A survey of coverage problems in wireless sensor networks," *Sensors & Transducers*, vol. 163, no. 1, p. 240, 2014.
- [3] S. Oktug, A. Khalilov, and H. Tezcan, "3d coverage analysis under heterogeneous deployment strategies in wireless sensor networks," in *2008 Fourth Advanced International Conference on Telecommunications*. IEEE, 2008, pp. 199–204.
- [4] W. Bentsz, "Dynamic coverage control and estimation in collaborative networks of human-aerial/space co-robots," Ph.D. dissertation, University of Michigan, Ann Arbor, 2020.
- [5] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

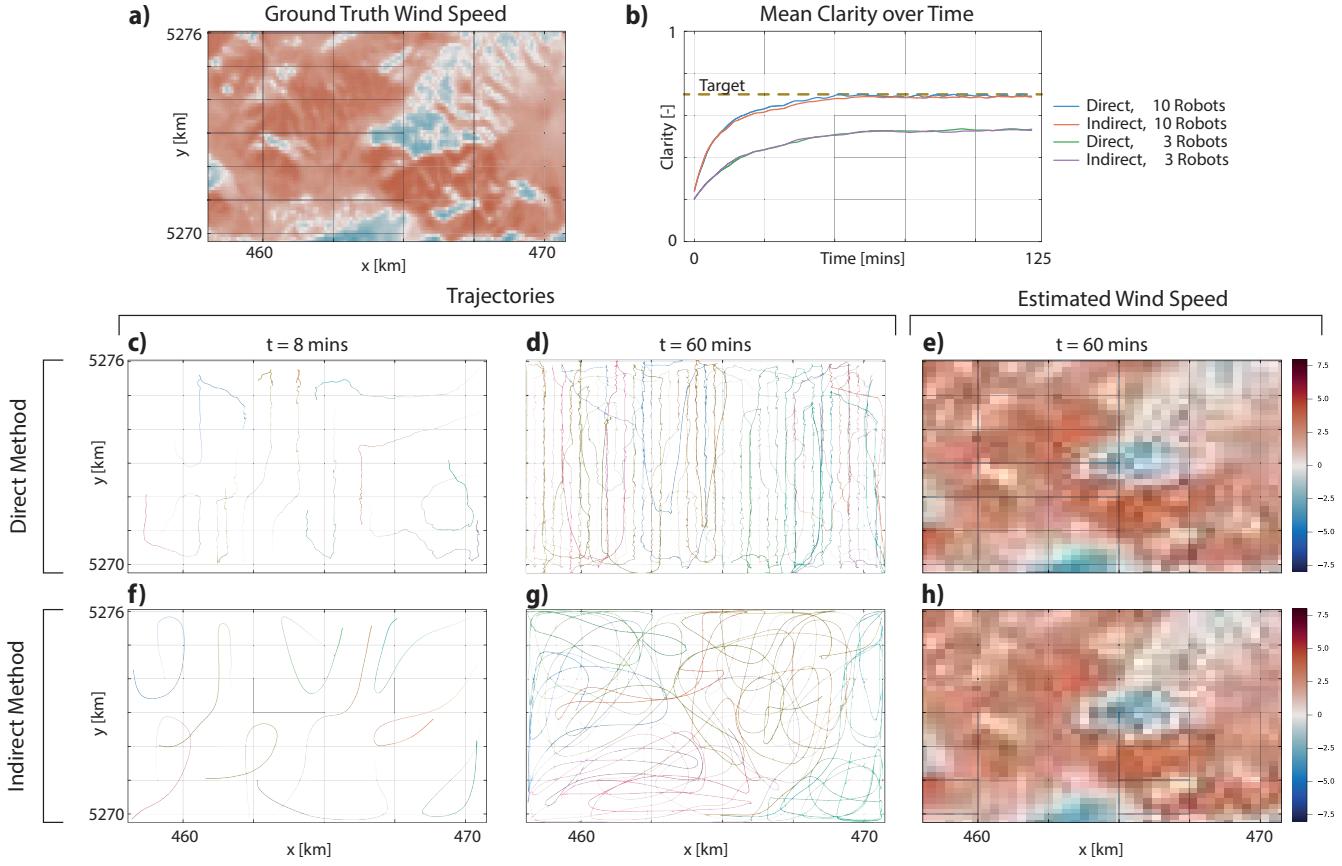


Fig. 2. Simulation results. (a) shows the ground truth wind speed at the end of the simulation. (b) shows the mean clarity against time. The mean is taken spatially. (c-e) show the behavior of the direct method. (f-h) show the behavior of the indirect method. (c, d, f, g) show the trajectories of the ten robots after eight minutes and after sixty minutes. (e, h) show the estimated wind speed, and it closely matches the ground truth in (a).

- [6] B. Moon, S. Chatterjee, and S. Scherer, “Tigris: An informed sampling-based algorithm for informative path planning,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2022, pp. 5760–5766.
- [7] A. Bry and N. Roy, “Rapidly-exploring random belief trees for motion planning under uncertainty,” in *IEEE International Conference on Robotics and Automation*, 2011, pp. 723–730.
- [8] N. Cao, K. H. Low, and J. M. Dolan, “Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms,” *arXiv preprint arXiv:1302.0723*, 2013.
- [9] C. Xiao and J. Wachs, “Nonmyopic informative path planning based on global kriging variance minimization,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1768–1775, 2022.
- [10] G. Mathew and I. Mezić, “Metrics for ergodicity and design of ergodic dynamics for multi-agent systems,” *Physica D: Nonlinear Phenomena*, vol. 240, no. 4-5, pp. 432–442, 2011.
- [11] L. Dressel and M. J. Kochenderfer, “Tutorial on the generation of ergodic trajectories with projection-based gradient descent,” *IET Cyber-Physical Systems: Theory & Applications*, vol. 4, no. 2, pp. 89–100, 2019.
- [12] D. R. Agrawal and D. Panagou, “Sensor-based planning and control for robotic systems: Introducing clarity and perceivability,” *IEEE Control Systems Letters*, 2023.
- [13] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.
- [14] S. Sarkka and J. Hartikainen, “Infinite-dimensional Kalman filtering approach to spatio-temporal Gaussian process regression,” in *Artificial intelligence and statistics*. PMLR, 2012, pp. 993–1001.
- [15] A. Küper and S. Waldherr, “Numerical Gaussian process Kalman filtering for spatiotemporal systems,” *IEEE Transactions on Automatic Control*, vol. 68, no. 5, pp. 3131–3138, 2022.
- [16] A. Carron, M. Todescato, R. Carli, L. Schenato, and G. Pillonetto, “Machine learning meets Kalman filtering,” in *55th IEEE Conference on Decision and Control*, 2016, pp. 4594–4599.
- [17] K. B. Naveed, D. Agrawal, C. Vermillion, and D. Panagou, “Eclares: Energy-aware clarity-driven ergodic search,” in *IEEE International Conference on Robotics and Automation*, 2024, pp. 14326–14332.
- [18] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 2006.
- [19] S. Särkkä and A. Solin, *Applied stochastic differential equations*. Cambridge University Press, 2019, vol. 10.
- [20] D. R. Agrawal and D. Panagou, “Multi-agent clarity-aware dynamic coverage with Gaussian processes,” *arXiv preprint arXiv:2403.17917*, 2024.
- [21] D. Dong, H. Berger, and I. Abraham, “Time optimal ergodic search,” *arXiv preprint arXiv:2305.11643*, 2023.
- [22] A. Gelb *et al.*, *Applied optimal estimation*. MIT press, 1974.
- [23] B. Haydon, K. D. Mishra, P. Keyantuo, D. Panagou, F. Chow, S. Moura, and C. Vermillion, “Dynamic coverage meets regret: Unifying two control performance measures for mobile agents in spatiotemporally varying environments,” in *60th IEEE Conference on Decision and Control*, 2021, pp. 521–526.
- [24] C. Schläger, G. Kirchengast, and J. Fuchsberger, “Generation of high-resolution wind fields from the wegenernet dense meteorological station network in southeastern austria,” *Weather and Forecasting*, vol. 32, no. 4, pp. 1301–1319, 2017.
- [25] H. Wackernagel, *Multivariate geostatistics: an introduction with applications*. Springer Science & Business Media, 2003.
- [26] R. B. Christianson, R. M. Polley, and R. B. Gramacy, “Traditional kriging versus modern Gaussian processes for large-scale mining data,” *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 16, no. 5, pp. 488–506, 2023.
- [27] E. L. Ince, *Ordinary differential equations*. Longmans, Green and Company Limited, 1927.

## APPENDIX

### A. Marginal and Conditional Distributions

Consider a random variable  $Z \in \mathbb{R}^{n+m}$ , given by

$$Z = \begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \right)$$

Then the marginal distributions are given by

$$\begin{aligned} X &\sim \mathcal{N}(\mu_x, \Sigma_{xx}) \\ Y &\sim \mathcal{N}(\mu_y, \Sigma_{yy}) \end{aligned}$$

and the conditional distributions are given by

$$\begin{aligned} (X|Y=y) &\sim \mathcal{N}(\mu, \Sigma), \\ \mu &= \mu_x + \Sigma_{xy} \Sigma_{yy}^{-1} (y - \mu_y) \\ \Sigma &= \Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx} \end{aligned}$$

Now consider two random variables  $X, Y$ , related by

$$\begin{aligned} X &\sim \mathcal{N}(\mu, P) \\ (Y|X=x) &\sim \mathcal{N}(Cx, R) \end{aligned}$$

where  $X \in \mathbb{R}^n$ ,  $Y \in \mathbb{R}^m$ ,  $C \in \mathbb{R}^{m \times n}$ ,  $P \in \mathbb{S}_{++}^n$ ,  $R \in \mathbb{S}_{++}^m$ .

What this means is that we have an observation model

$$y = Cx + w, \quad w \sim \mathcal{N}(0, R)$$

### B. Gaussian Processes

The kernel of a GP is defined by the following property

**Definition 3.** The *kernel function* of a Gaussian Process  $Z \sim \text{GP}(m(x), k(x, x'))$  with mean function  $m : \mathbb{R}^d \rightarrow \mathbb{R}$  and kernel function  $k : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$  is defined as

$$k(x, x') = \mathbb{E}[(Z(x) - m(x))(Z(x') - m(x'))].$$

*Example 3.* The  $\nu$ -th order *Matern kernel* is given by

$$k_\nu(x_1, x_2) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left( \sqrt{2\nu} \lambda d \right)^\nu K_\nu \left( \sqrt{2\nu} \lambda d \right),$$

where  $\Gamma$  is the gamma function,  $K_\nu$  is the modified Bessel function of the second kind,  $d = \|x_1 - x_2\|$ , and  $\sigma, \lambda > 0$  are parameters of the kernel. The half-integer Matern kernels are given by

$$k_{1/2}(x_1, x_2) = \sigma^2 \exp(-\lambda d)$$

$$k_{3/2}(x_1, x_2) = \sigma^2 \left( 1 + \sqrt{3} \lambda d \right) \exp \left( -\sqrt{3} \lambda d \right)$$

$$k_{5/2}(x_1, x_2) = \sigma^2 \left( 1 + \sqrt{5} \lambda d + (5/3) \lambda^2 d^2 \right) \exp \left( -\sqrt{5} \lambda d \right)$$

where  $d = \|x_1 - x_2\|$ , and  $\sigma, \lambda > 0$  are hyperparameters of the kernel.

### C. Variograms

This section establishes a method to determine the hyperparameters of a Gaussian Process using an Empirical Variogram. This method is significantly more computationally efficient and accurate than standard methods of minimizing the marginal log-likelihood but is only suitable for isotropic kernels.

Consider the data set  $D = \{(x_i, y_i)\}_{i=1}^N$ , where  $x_i \in \mathbb{R}^d$ ,  $y_i \in \mathbb{R}$ . The goal is to determine the parameters of an isotropic kernel  $k : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$  that best fits the data.

**Definition 4.** [25, Eq 7.6] The *theoretical variogram* of a stationary random field  $Z : \mathbb{R}^d \rightarrow \mathbb{R}$  with zero mean is

$$\gamma(d) = \frac{1}{2} \mathbb{E} \left[ (Z(x') - Z(x))^2 \right],$$

where  $\|x' - x\| = d$ . The expectation is taken over  $x, x' \in \mathbb{R}^d$ .

The theoretical variogram is related to GP kernels as follows:

**Lemma 1.** Suppose  $Z$  is a zero-mean and isotropic Gaussian Process. Then the kernel  $k : \mathbb{R} \rightarrow \mathbb{R}$  and the theoretical variogram  $\gamma : \mathbb{R} \rightarrow \mathbb{R}$  are related by

$$\gamma(d) = k(0) - k(d)$$

*Proof.* For brevity, let  $f_1 = f(x_1)$ ,  $f_2 = f(x_2)$ . By the definition of the kernel, for a zero-mean GP

$$\begin{aligned} k(x_1, x_2) &= \mathbb{E}[(f(x_1) - m(x_1))(f(x_2) - m(x_2))] \\ &= \mathbb{E}[f_1 f_2] \end{aligned}$$

Similarly, from the definition of the theoretical variogram,

$$\begin{aligned} \gamma(d) &= \frac{1}{2} \mathbb{E}[(f(x_1) - f(x_2))^2] \\ &= \frac{1}{2} \mathbb{E}[f_1^2] - \mathbb{E}[f_1 f_2] + \frac{1}{2} \mathbb{E}[f_2^2] \\ &= \frac{1}{2} (E[f_1^2] + E[f_2^2]) - E[f_1 f_2] \\ &= \frac{1}{2} (k(x_1, x_1) + k(x_2, x_2)) - k(x_1, x_2) \\ &= \frac{1}{2} (2k(0)) - k(d) \end{aligned}$$

using  $d = x_2 - x_1$ . Therefore,

$$\gamma(d) = k(0) - k(d).$$

□

**Corollary 2.** In the spatiotemporal case, if the kernel is

$$k(t, x, t', x') = k_t(t, t') k_s(x, x') \quad (29)$$

the theoretical variogram is

$$\gamma(d_t, d_s) = k_t(0) k_s(0) - k_t(d_t) k_s(d_s) \quad (30)$$

We can use this Lemma to determine the parameters of the kernel. In particular, consider the empirical variogram:

**Definition 5.** The *empirical semi-variogram* given data  $D$  is  $\gamma : \mathbb{R} \rightarrow \mathbb{R}$ ,

$$\gamma(d) = \frac{1}{2|N(d)|} \sum_{N(d)} (y_i - y_j)^2$$

where  $N(d) \subset \mathbb{Z} \times \mathbb{Z}$  is the set of pairs  $(i, j)$  such that  $\|x_i - x_j\| \in (d - \epsilon, d + \epsilon)$  for some  $\epsilon > 0$ .

Then, given data  $D$ , we construct the empirical variogram.

For a given kernel  $k$  with hyperparameters  $\theta$ , we can compute the corresponding theoretical variogram, and use least-squares fitting to determine the set of hyperparameters  $\theta$  that best fit the data  $D$ .

#### D. Gaussian Processes to Stochastic Differential Equations

This section explains the equivalence between GP and SDE for a class of kernel functions. In this section, we focus on scalar GPs with zero mean,

$$f(t) \sim \text{GP}(0, k(t, t')),$$

where  $k : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$  is denoted with  $t$  to remind the reader that we consider a single (i.e. temporal) dimension.

We use the following convention of a Fourier Transform<sup>6</sup> of a function  $g : \mathbb{R} \rightarrow \mathbb{R}$ :

**Definition 6.** The Fourier Transform of a function  $g : \mathbb{R} \rightarrow \mathbb{R}$  is the function  $G : \mathbb{R} \rightarrow \mathbb{R}$ ,

$$G(\omega) = \mathcal{F}[g](\omega) = \int_{-\infty}^{\infty} g(t) e^{-i\omega t} dt$$

The *Inverse Fourier Transform* is

$$g(t) = \mathcal{F}^{-1}[G](t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} G(\omega) e^{i\omega t} d\omega$$

This convention has the following properties:

$$\mathcal{F}\left[\frac{d^n g}{dt^n}\right](\omega) = (i\omega)^n \mathcal{F}[g](\omega)$$

The Wiener-Khinchin theorem relates the kernel function to the power spectrum of a stochastic process:

$$S(\omega) = \mathcal{F}[k](\omega)$$

here, we write  $k(\tau) = k(t, t')$  for any  $|t - t'| = \tau$ . When  $S$  is a rational function of even order  $2n_k$ , we can decompose  $S$  as

$$S(\omega) = L(\omega) L(-\omega)$$

where

$$L(\omega) = \frac{b_{n_k-1}(i\omega)^{n_k-1} + b_{n_k-2}(i\omega)^{n_k-2} + \dots + b_0}{(i\omega)^{n_k} + a_{n_k-1}(i\omega)^{n_k-1} + \dots + a_0}$$

Given this decomposition, we know that the stochastic process  $f$  is a realization of a white noise process  $W(t)$  that has been colored using the transfer function  $L(\omega)$ . Therefore, the state-space model of the system is

$$ds = Asdt + BdW$$

$$z = Cs$$

where  $s \in \mathbb{R}^{n_k}$  is the state,  $W(t)$  is the standard (1D) white noise process. The output  $z(t)$  will have the correct kernel function. Here, the constant matrices  $A \in \mathbb{R}^{n_k \times n_k}$ ,  $B \in$

<sup>6</sup>In Mathematica, one must specify `FourierParameters -> {1, -1}` to yield the correct convention.

$\mathbb{R}^{n_k \times 1}$ ,  $C \in \mathbb{R}^{1 \times n_k}$  are

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ & & \ddots & & \\ 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n_k-1} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$C = [b_0 \ b_1 \ b_2 \ \dots \ b_{n_k-1}]$$

To create a realization of  $f$  that has the desired kernel function, simulate the SDE starting from  $s_0 \sim N(0, \Sigma_0)$ , where  $\Sigma_0 \in \mathbb{R}^{n_k \times n_k}$  is the solution to the Lyapunov equation  $AX + XA^T + BB^T = 0$ .

Finally, the discrete time version of this, with a sampling period  $\Delta t$  is

$$s_{k+1} = \Phi s_k + w_k, \quad w_k \sim \mathcal{N}(0, W)$$

$$z_k = Cs_k$$

where

$$\Phi = e^{A\Delta t}$$

$$W = \int_0^{\Delta t} e^{A\tau} BB^T e^{A^T\tau} d\tau$$

Some analytic expressions are derived below.

*Example 4* (Matern 1/2). The 1D Matern-1/2 kernel is

$$k_{1/2}(d) = \sigma^2 \exp(-\lambda d)$$

It has a power-spectral density

$$S_{1/2}(\omega) = \frac{2\lambda\sigma^2}{\lambda^2 + \omega^2}$$

and rational decomposition

$$L_{1/2}(\omega) = \frac{\sigma\sqrt{2\lambda}}{(i\omega) + \lambda}$$

Therefore, the state-space representation is

$$\left( \begin{array}{c|c} A & B \\ \hline C & \end{array} \right) = \left( \begin{array}{c|c} -\lambda & 1 \\ \hline \sigma\sqrt{2\lambda} & \end{array} \right)$$

*Example 5* (Matern 3/2). The 1D Matern-3/2 kernel is

$$k_{3/2}(x_1, x_2) = \sigma^2 \left( 1 + \sqrt{3}\lambda d \right) \exp\left(-\sqrt{3}\lambda d\right)$$

It has a power-spectral density

$$S_{3/2}(\omega) = \frac{12\sqrt{3}\lambda^3\sigma^2}{(3\lambda^2 + \omega^2)^2}$$

and rational decomposition

$$L_{3/2}(\omega) = \frac{\sqrt{12\sqrt{3}\lambda^{3/2}}\sigma}{(i\omega)^2 + 2\sqrt{3}\lambda(i\omega) + 3\lambda^2}$$

Therefore, the state-space representation is

$$\left( \begin{array}{c|c} A & B \\ \hline C & \end{array} \right) = \left( \begin{array}{cc|c} 0 & 1 & 0 \\ -3\lambda^2 & -2\sqrt{3}\lambda & 1 \\ \hline \sqrt{12\sqrt{3}\lambda^{3/2}}\sigma & 0 & \end{array} \right)$$

*Example 6* (Matern 5/2). The 1D Matern-5/2 kernel is

$$k_{5/2}(x_1, x_2) = \sigma^2 \left( 1 + \sqrt{5}\lambda d + (5/3)\lambda^2 d^2 \right) \exp \left( -\sqrt{5}\lambda d \right)$$

It has a power-spectral density

$$S_{5/2}(\omega) = \sigma^2 \frac{400\sqrt{5}\lambda^5}{3(5\lambda^2 + \omega^2)^3}$$

and rational decomposition

$$L_{5/2}(\omega) = \frac{\sqrt{\frac{400\sqrt{5}}{3}}\lambda^{5/2}\sigma}{(i\omega)^3 + 3\sqrt{5}\lambda(i\omega)^2 + 5\sqrt{5}\lambda^3}$$

Therefore, the state-space representation is

$$\left( \begin{array}{c|c} A & B \\ \hline C & \end{array} \right) = \left( \begin{array}{ccc|c} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -5\sqrt{5}\lambda^3 & -15\lambda^2 & -3\sqrt{5}\lambda & 1 \\ \hline \sqrt{\frac{400\sqrt{5}}{3}}\lambda^{5/2}\sigma & 0 & 0 & \end{array} \right)$$

#### E. Solutions to the Riccati Equation

**Lemma 3.** Consider a differential equation

$$y'(t) = -\alpha y(t)^2 - \beta y(t) - \gamma \quad (31a)$$

$$y(0) = y_0 \quad (31b)$$

where  $\alpha, \beta, \gamma \in \mathbb{R}$ ,  $\alpha \neq 0$ , and  $\delta^2 = \beta^2 - 4\alpha\gamma > 0$ . Then, the solution is given by

$$y(t) = \frac{1}{2\alpha} \left( -\beta + \delta + \frac{2\delta\rho_0}{(2\delta + \rho_0)e^{\delta t} - \rho_0} \right) \quad (32)$$

where  $\rho_0 = \beta - \delta + 2\alpha y_0$ .

*Proof.* This is a second-order nonlinear differential equation, also known as the scalar Riccati equation. As proposed in [27, Ch. 2.15], consider the substitution

$$y(t) = \frac{u'(t)}{\alpha u(t)} \quad (33)$$

Then, it is equivalent to the following differential equation

$$u''(t) = -\beta u'(t) - \alpha\gamma u(t) \quad (34a)$$

$$y_0 = \frac{u'(0)}{\alpha u(0)} \quad (34b)$$

This linear second-order differential equation has a unique solution

$$u(t) = (c_2 e^{\delta t} + c_1) e^{-\frac{1}{2}t(\beta+\delta)} \quad (35)$$

where  $\delta = \sqrt{\beta - 4\alpha\gamma}$  and the constants  $c_1, c_2$  depend on the boundary condition. Evaluating the boundary conditions, we have the relationship

$$y_0 = \frac{\frac{1}{2}(c_1 + c_2)(-\beta - \delta) + c_2\delta}{\alpha(c_1 + c_2)} \quad (36)$$

Evaluating  $y = u' / (\alpha u)$ , we have

$$y(t) = \frac{1}{2\alpha} \left( -\beta + \delta - \frac{2c_1\delta}{c_2 e^{\delta t} + c_1} \right) \quad (37)$$

Plugging in the boundary condition, we arrive at

$$y(t) = \frac{1}{2\alpha} \left( -\beta + \delta + \frac{2\delta\rho_0}{(2\delta + \rho_0)e^{\delta t} - \rho_0} \right) \quad (38)$$

where  $\rho_0 = \beta - \delta + 2\alpha y_0$ , independent of  $c_1, c_2$ .  $\square$

**Corollary 4.** The limiting value of (32) is

$$y_\infty = \lim_{t \rightarrow \infty} y(t) = \frac{\delta - \beta}{2\alpha}. \quad (39)$$

**Corollary 5.** The inverse of (32) is given by

$$t = \frac{1}{\delta} \log \left( \frac{\rho_0 (2\delta + \rho_f)}{\rho_f (2\delta + \rho_0)} \right) \quad (40)$$

where  $\delta^2 = \beta^2 - 4\alpha\gamma$ ,  $\rho_0 = \beta - \delta + 2\alpha y_0$  and  $\rho_f = \beta - \delta + 2\alpha y_f$ .