AIAA SciTech Forum
6-10 January 2025, Orlando, FL
AIAA SCITECH 2025 Forum

10.2514/6.2025-2069

Reconfiguration Costs in Coupled Sensor Configuration and Path-Planning for Dynamic Environments

Prakash Poudel* and Raghvendra V. Cowlagi†
Worcester Polytechnic Institute, Worcester, MA, 01609, USA

This paper addresses the problem of sensor placement to assist a mobile vehicle that navigates in an unknown environment. We refer this spatially and temporally varying environment as a threat field. The vehicle is required to navigate with a minimum exposure to this field. We consider a sensor network separate from the mobile vehicle that provides a pointwise noisy measurements of the threat field. We propose an iterative method, referred as coupled sensor configuration and path-planning (CSCP) to solve this problem. At each iteration, the optimal sensor configuration is determined by maximizing a metric that includes uncertainty reduction in a path cost and minimization of sensor movement. A context-relevant mutual information (CRMI) metric enables sensor placement in locations that reduce uncertainty in the path cost, rather than in the environment state. Sensor reconfiguration cost is computed based on the distance traveled by the sensors from a current set of locations to new locations. To address the problem of combinatorial growth in the feasible sensor configurations with increasing number of sensors, we execute greedy optimization of placing one sensor at a time. Once the new sensor locations are identified, the algorithm updates the threat estimate with new measurements, and recalculates the path with minimum expected exposure to the threat. We perform numerical simulations to show the effectiveness of the proposed method, and conduct comparative study between the CSCP method with and without incorporating the sensor reconfiguration cost.

I. Introduction

Path-planning and trajectory optimization methods for uncrewed aerial vehicles (UAVs) is an area with a rich literature, summarized in the handbook [1], for a large variety of applications. Methods for waypoint navigation, obstacle avoidance in various kinds of environments (known, partially known, unknown, GPS-denied, indoors, etc.), minimum-time travel, multi-UAV formation flight, and cooperative planning for surveillance coverage are widely reported. Regardless of the application, path-planning methods typically use geometric methods based on discretization of the environment, whereas trajectory optimization methods may use indirect variational necessary conditions of optimality or direct transcription based on appropriate parametrization.

We are interested in the problem of path-planning with minimum exposure to a spatially and temporally varying scalar field, which we refer to as the *threat field.* The threat field may signify unfavorable attributes of the environment, such as hazards, or the negative / inverse of favorable attributes. For example, the threat field may indicate the risk of attack by an adversary, which we would like to minimize during the entirety flight of the UAV. For brevity, we will refer to this problem as the *minimum-threat path-planning* problem for the *ego vehicle.*

Modern architectures of networked and distributed autonomy leverage heterogeneous agents acting in cooperation. In the context of the minimum-threat problem, we may envision a distributed sensor network that collects and process data about the threat field. The ego vehicle can access this knowledge of the threat field to inform its path-planning. More importantly, consider an architecture where the sensor agents are mobile (e.g., surveillance UAVs) and move as commanded. Such a networked autonomy architecture naturally leads to the problem of sensor placement or, more generally, sensor configuration.

The placement problem becomes somewhat trivial if there a very large number of mobile sensor agents. In this case,

*Graduate Research Assistant, Aerospace Engineering Department. AIAA Student Member.
†Associate Professor, Aerospace Engineering Department. AIAA Senior Member.

a simple uniform spatial distribution of the agents would suffice. However, in practice the sensor network is constrained in the number of agents available and their energy usage. Consider, for example, a sensor network of unmanned aerial vehicles (UAVs) for surveillance of a threat field like wildfire in a large area. Due to cost and battery limitations, it may not be possible to achieve full area coverage quickly enough to inform the actions of a ground robot to safely navigate the environment. This situation exemplifies the broader problem of path-planning with a *minimal* number of sensor measurements, and in turn, highlights the need for optimal sensor configuration *in the context* of, i.e., *coupled* with path-planning. In this paper, we focus on the problem of optimally placing and moving the mobile sensor agents to collect threat data of the most relevance to the minimum-threat path-planning.

Related Work: The proposed approach naturally draws on the wealth of knowledge in optimal estimation theory [2–4] based on the Kalman filter and its many variants. For systems with nonlinear dynamics and/or non-Gaussian noise characteristics, commonly-used approaches include the extended Kalman filter [4], the linearization-free unscented Kalman filter [5], the ensemble Kalman filter [6] and the particle filter [7, 8].

Path- and motion-planning are similarly mature areas of research. Generally, path-planning under uncertainty involves finding paths that minimize the expected cost. Classical approaches to path-planning include cell decomposition, probabilistic roadmaps, and artificial potential field techniques [9], [10]. Dijkstra's algorithm, A*, and their variants are branch-and-bound optimization algorithms that leverage heuristics to effectively steer the path search towards the goal. Modern approaches to path planning leverage advanced methodologies such as adaptive informative path-planning [15], coverage path-planning [16] and informed sampling-based path planning [17]. More recently, techniques based on reinforcement learning [11–13] and fuzzy logic [14] are reported.

Sensor configuration techniques including placement, tasking, and scheduling are reported in recent years. Among these, information theoretic techniques achieve configure sensors by maximizing metrics such as the determinant or largest eignevalue of the Fisher information matrix (FIM) [18, 19], entropy [20], Kullback-Leibler (KL) divergence [21], mutual information [22–24], and frame potential [25]. Broadly speaking, these metrics quantify the reduction in uncertainty in the knowledge of the environment state. Most of these information-maximizing sensor configuration approaches employ greedy optimization techniques for mitigating the curse of dimensionality [25–27].

Another example of information-theoretic metric [28] utilizes two metrics, one associated with mutual information based on objection detection, and another with mutual information based on classification of the detected objects. More recently, a reinforcement learning-based method for coupled sensor placement and path-planning [29] learns a so-called proximal policy. A bilevel optimization technique to optimally place sensors for estimating the emission rates of multiple sources using is reported in [30]. Sensor placement in a continuous spatial domain for nonlinear distributed parameter systems using Lagrange polynomials and the orthogonal collocation method is reported [31].

Other data-driven sensor placement methods for various applications, namely signal reconstruction [32], and target tracking are also reported [33]. Machine learning-based sensor placement techniques are reported for efficient estimation with a minimal number of measurements [34, 35].

These sensor configuration works largely ignore the context, i.e., focus on optimally estimating the state of the environment without context to a specific objective such as path-planning. Prior works by the second author and co-workers address the coupling between sensor placement and path-planningfor static environments. A heuristic task-driven sensor placement approach called the interactive planning and sensing (IPAS) for static environments is reported in [36]. The IPAS method is shown to outperform several decoupled sensor placement methods in terms of the total number of measurements needed to achieve near-optimal paths. Sensor configuration for location and field-of-view is reported in [37], also for static fields. Sensor placement for multi-agent path-planning based on entropy reduction is reported in [38]. Recent work by the authors presents a coupled method based on maximizing a metric called context-relevant mutual information (CRMI) [39]. CRMI is the mutual information of the measurement and the path cost, rather than the measurement and the environment state (which is typically used).

Although it is commonly studied in the mobile sensor network literature, the problem of accounting for sensor reconfiguration costs is relatively less studied for sensor placement. Reconfiguration becomes an important especially when multiple iterations of sensor configuration and estimation are conducted, which in turn may be necessary when

the number of sensors is small. Some examples include consideration of reconfiguration cost of the sensor network topology [40], or the total energy consumption of the sensor network [41, 42].

Contributions: The contribution of this work to the literature is that we present a coupled sensor configuration and path-planning (CSCP) method that includes a sensor reconfiguration cost. We develop an iterative algorithm for CSCP. At each iteration, a threat estimate is first computed using sensor measurements. Next, a path-planning algorithm finds a path of minimum expected threat. Next, optimal sensor placements are computed to maximize a metric that collectively maximize the path-dependent CRMI and minimize the sensor movement, and the iterations repeat. We compare the proposed CSCP method with our previous CSCP method that ignores the sensor reconfiguration cost.

II. Problem Formulation

For the sake of consistency across our different publications, we reproduce the problem formulation from our most recent work [39].

Let $\mathbb{R}$ represent the set of real numbers, and $\mathbb{N}$ the set of natural numbers. For any $N \in \mathbb{N}$, we denote by $[N]$ the set $\{1, 2, \ldots, N\}$, and by $\boldsymbol{I}_{(N)}$ the identity matrix of size N.

Consider a closed square region denoted by $\mathcal{W} \subset \mathbb{R}^2$ and referred as the *workspace*, within which the mobile agent and sensors operate. In this workspace, consider a grid consisting of N_g uniformly spaced points. The coordinates of these points in a prespecified Cartesian coordinate axis system are denoted by $\boldsymbol{x}_i$, for each $i \in N_\mathrm{g}$. The distance between the adjacent grid points is denoted by δ. The mobile agent traverses grid points according to the "4-way adjacency rule", such that the adjacent points are top, down, left, and right. We formulate the path planning problem for a actor as a graph search problem on a graph, $\mathcal{G} = (V, E)$ with $V = [N_\mathrm{g}]$ such that each vertex in V is uniquely associated with a grid point. The set of edges E in this graph consists of pairs of grid points that are geometrically adjacent to each other.

A *threat field*, denoted as $c : \mathcal{W} \times \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{>0}$, is a time-varying scalar field that takes strictly positive values, indicating regions with higher intensity that are potentially hazardous and unfavorable. A path between two prespecified initial and goal vertices, $i_s, i_g \in V$, is defined as a finite sequence $\boldsymbol{v} = \{v_0, v_1, \ldots, v_L\}$ of successively adjacent vertices. This sequence starts at the initial vertex $v_0 = i_s$ and ends at the goal vertex $v_L = i_g$, where $L \in \mathbb{N}$ represents the number of vertices in the sequence. The edge transition costs, which account for the expenses incurred when an actor moves between vertices in a graph, are determined by a scalar function $g : E \to \mathbb{R}_{>0}$. This function assigns a value to each edge in the graph, representing the associated cost or effort required for traversal and is defined as,

$$g((i, j), t) = c(\boldsymbol{x}_j, t), \text{ for } \quad i, j \in [N_\mathrm{g}], \quad (i, j) \in E \tag{1}$$

The cost $J(\boldsymbol{v}) \in \mathbb{R}_{>0}$ indicates the total threat exposure for an actor on its traversal along a path $\boldsymbol{v}$ and is defined as the sum of edge transition costs, $J(\boldsymbol{v}) = \sum_{\ell=1}^{L} g((v_{\ell-1}, v_\ell), \ell\Delta t_s, \delta)$. The main problem of interest is to find a path with a minimum cost, $\boldsymbol{v}^*$. Since the threat field is unknown and is changing dynamically, estimation of the threat field in the environment is essential. A network of N_s sensors, where $N_\mathrm{s} \ll N_\mathrm{g}$, can be used to measure the intensity of threat. These sensor measurements are denoted $\boldsymbol{z}(\boldsymbol{x}, t; \boldsymbol{q}) = \{z_1(\boldsymbol{x}, t; \boldsymbol{q}), z_2(\boldsymbol{x}, t; \boldsymbol{q}), \ldots, z_{N_s}(\boldsymbol{x}, t; \boldsymbol{q})\}$ will be used to define the filter required for estimating the state of the dynamic system. Sensors are placed at distinct grid points, and the set of this grid points is called the *sensor configuration*, $\boldsymbol{q} = \{q_1, q_2, \ldots, q_{N_s}\} \subset [N_\mathrm{g}]$.

The threat field is modeled in parametric form as $c(\boldsymbol{x}, t) := 1 + \sum_{n=1}^{N_P} \theta_n(t)\phi_n(\boldsymbol{x}) = 1 + \boldsymbol{\Phi}(\boldsymbol{x})^\intercal \boldsymbol{\Theta}(t)$, with $\boldsymbol{\Phi}(\boldsymbol{x}) := [\phi_1(\boldsymbol{x}) \ldots \phi_{N_P}(\boldsymbol{x})]^\intercal$, and $\phi_n(\boldsymbol{x}) := \exp(-(\boldsymbol{x} - \overline{\boldsymbol{x}}_n)^\intercal (\boldsymbol{x} - \overline{\boldsymbol{x}}_n)/2a_n)$ representing the basis functions for each $n \in [N_P]$. Here, N_P represents the number of threat parameters involved to define the threat field. The values of the constants $a_n \in \mathbb{R}_{>0}$ and $\overline{\boldsymbol{x}}_n \in \mathcal{W}$ are prespecified and chosen in such a manner that the combined interiors of the significant support regions cover the entire workspace [36]. The parameter $\boldsymbol{\Theta}(t) := [\theta_1(t) \ldots \theta_{N_P}(t)]^\intercal$ is to be estimated. The temporal evolution of the threat is modeled by

$$\dot{\boldsymbol{\Theta}}(t) = A_c \boldsymbol{\Theta}(t) + \boldsymbol{\omega}(t), \tag{2}$$

where $\omega(t) \sim \mathcal{N}(0, Q_c)$ is white process noise with $Q_c := \sigma_P \boldsymbol{I}_{(N_P)}$. We restrict our attention to linear threat evolution models for now, but it will become clear that the proposed method can be easily extended to nonlinear models as well.

The matrix A_c represents the evolution of threat parameters $\boldsymbol{\Theta}(t)$. This evolution model may potentially - but not necessarily - be derived from an underlying physical model of the threat. For example, the solution to a heat diffusion equation, $\frac{\partial c}{\partial t} = \alpha(\frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2})$ can be approximated by $c(\boldsymbol{x}, t) = 1 + \boldsymbol{\Phi}(\boldsymbol{x})^\intercal \boldsymbol{\Theta}(t)$. It can be shown that the parameters $\boldsymbol{\Theta}(t)$ satisfy $\dot{\boldsymbol{\Theta}}(t) = \alpha \frac{\boldsymbol{\Phi}^\intercal}{|\boldsymbol{\Phi}|^2} \nabla^2 \boldsymbol{\Phi} \boldsymbol{\Theta}(t)$, such that $A_c = \alpha \frac{\boldsymbol{\Phi}^\intercal}{|\boldsymbol{\Phi}|^2} \nabla^2 \boldsymbol{\Phi}$.

The model (2) can be easily discretized to form a difference equation of the form

$$\boldsymbol{\Theta}_k = A\boldsymbol{\Theta}_{k-1} + \boldsymbol{\omega}_{k-1}, \tag{3}$$

where $A := \boldsymbol{I}_{(N_P)} + A_c \Delta t + \frac{(A_c)^2 (\Delta t)^2}{2!}$ The process noise is zero-mean, white, and Gaussian, i.e., $\omega(k) \sim \mathcal{N}(0, Q)$, where $Q := Q_c \Delta t + \frac{(A_c Q_c + Q_c A_c^\intercal)(\Delta t)^2}{2!}$ [4]. In our implementation, we ignore higher-order terms of Δt beyond the linear term.

The measurements obtained from each sensor are modeled by $z_k := c(\boldsymbol{x}_{q_k}, t) + \boldsymbol{\eta}_k = H_k(\boldsymbol{q})\boldsymbol{\Theta}_k + \boldsymbol{\eta}_k$, where

$$H_k(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{q}_{k,1}}) & \boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{q}_{k,2}}) & \dots & \boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{q}_{k,N_s}}) \end{bmatrix}^\intercal,$$

and $\boldsymbol{\eta}_k \sim \mathcal{N}(0, R)$ is zero mean measurement noise with covariance $R \succ 0$.

The threat parameters $\boldsymbol{\Theta}(t)$ are unknown quantities, and therefore we generate stochastic estimates with mean value $\widehat{\boldsymbol{\Theta}}(t)$ and estimation error covariance P. For any path, $\boldsymbol{v} = \{v_0, v_1, \dots, v_L\}$ in $\mathcal{G}$, the cost of the path is

$$J(\boldsymbol{v}) := L + \delta \textstyle\sum_{l=1}^{L} \boldsymbol{\Phi}(\boldsymbol{x}_l)^\intercal \boldsymbol{\Theta}_l.$$

The cost J becomes a random variable with distribution dependent on the estimate of $\boldsymbol{\Theta}$. The main problem of interest is then formulated as follows.

Problem 1 *For a prespecified termination threshold, $\epsilon > 0$ and some finite iterations $k = 0, 1, \dots, M$, find a sequence of sensor configurations $\boldsymbol{q}_k^*$ and a path $\boldsymbol{v}^*$ with minimum expected cost $\widehat{J}(\boldsymbol{v}^*)$ and such that $Var[J(\boldsymbol{v}^*)] \leqslant \epsilon$.*

III. Coupled Sensing and Planning

Coupled sensor configuration and path-planning (CSCP) is an iterative approach to solve Problem 1. An illustration of the various components of the CSCP method and their interactions is shown in Fig. 1. Based on the threat field model and a sensor network, we have a discrete predictive and measurement model as discussed in Sec. II. At each iteration, a sensor configuration is determined, and measurements of the threat field are collected. The optimal sensor configuration is found by maximizing an objective function based on a *context-relevant mutual information* (CRMI) and a sensor reconfiguration cost. Next, these sensor measurements are used to update the threat field estimates in an estimator. To easily quantify the uncertainty in path cost, we prefer an estimator that maintains the mean and covariance of the threat parameter estimate. To this end, the Unscented Kalman Filter (UKF) is chosen for its ease of use with nonlinear systems, although the examples currently discussed in this paper are limited to linear threat evolution models. Next, the path is planned based on the new threat field estimate, and this process continues until the path cost variance is reduced below a prespecified threshold ϵ. We implement Dijkastra's algorithm to plan the path.

A. Mutual Information between Path Cost and Measurement (CRMI)

The context-relevant mutual information (CRMI) measures the information shared between the path cost and the measurements in a path-planning problem. It focuses on the spatial locations that are relevant to the path planning, neglecting those that are distant from the planned path. In essence, it captures the most pertinent information for the

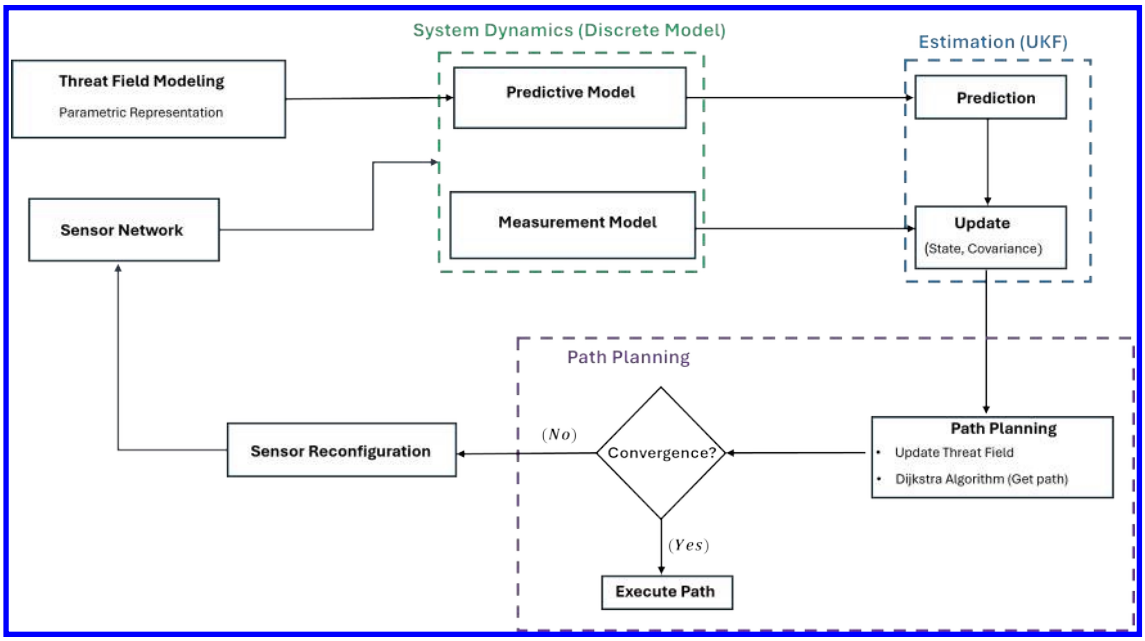


Fig. 1 Schematic illustration of the proposed CSCP method.

concurrent path-planning task by considering the proximity of locations to the planned path.

For any path $\boldsymbol{v}$, the expected cost is $\widehat{J}(\boldsymbol{v}) := L + \delta \sum_{l=1}^{L} \boldsymbol{\Phi}(\boldsymbol{x}_l)^\intercal \widehat{\boldsymbol{\Theta}}_l$. Considering the Gaussian PDFs, the joint PDF $p(J_k, \boldsymbol{z}_k)$ of the path cost and measurement variables is

$$p(J_k, \boldsymbol{z}_k) = \mathcal{N}\left(\begin{bmatrix} J_k \\ \boldsymbol{z}_k \end{bmatrix} : \begin{bmatrix} \widehat{J}_{k|k-1} \\ \widehat{\boldsymbol{z}}_k \end{bmatrix}, \begin{bmatrix} P_{JJ_{k|k-1}} & P_{J\boldsymbol{z}_{k|k-1}} \\ P^\intercal_{J\boldsymbol{z}_{k|k-1}} & P_{\boldsymbol{z}\boldsymbol{z}_{k|k-1}} \end{bmatrix}\right).$$

The variance of the path cost is

$$\begin{aligned} P_{JJ_{k|k-1}} &:= \mathbb{E}\left[\left(J(\boldsymbol{v}) - \widehat{J}(\boldsymbol{v})\right)^2\right] = \mathbb{E}\left[\left(\delta \sum_{l=1}^{L} \boldsymbol{\Phi}^\intercal(\boldsymbol{x}_{\boldsymbol{v}_l})\left(\boldsymbol{\Theta}_l - \widehat{\boldsymbol{\Theta}}_l\right)\right)^2\right], \\ &= \delta^2 \sum_{l=1}^{L} \left(\boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{v}_l})^\intercal P_{k_l} \boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{v}_l})\right) + 2\delta^2 \sum_{l<m,\ l,m\in[L]}^{L} \left(\boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{v}_l})^\intercal P_{k_{lm}} \boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{v}_m})\right). \end{aligned}$$

To compute $P_{JJ_{k|k-1}}$, we need to find $\boldsymbol{\Phi}$ and the error covariance P for each grid point $\boldsymbol{v}_l$ along the path. This involves propagating the parameter uncertainty over time intervals required to traverse between grid points, determining P_{k_l} and $P_{k_{lm}}$. The covariance of the measurement and the cross covariance between the path cost and the measurement random vector are formulated as:

$$P_{J\boldsymbol{z}_{k|k-1}} = \mathbb{E}\left[(\boldsymbol{z} - \widehat{\boldsymbol{z}})\left(J(\boldsymbol{v}) - \widehat{J}(\boldsymbol{v})\right)\right] = \delta \sum_{l=1}^{L} \left(\boldsymbol{\Phi}(\boldsymbol{x}_{\boldsymbol{v}_l})^\intercal P_{k_l}\right) H_k^\intercal(\boldsymbol{q}), \tag{4}$$

$$P_{\boldsymbol{z}\boldsymbol{z}_{k|k-1}} = H_k(\boldsymbol{q}) P_{\boldsymbol{\Theta}\boldsymbol{\Theta}_{k|k-1}} H_k^\intercal(\boldsymbol{q}) + R_k. \tag{5}$$

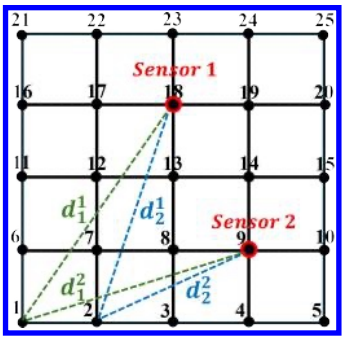


Fig. 2 Illustration of sensor reconfiguration cost.

Here, $P_{\boldsymbol{\Theta}\boldsymbol{\Theta}_{k|k-1}}$ is the priori state covariance that is obtained from the UKF algorithm. Finally, the CRMI is calculated as

$$I(J_k; z_k(\boldsymbol{q})) = \frac{1}{2}\log\left(\frac{|P_{JJ_{k|k-1}}|}{|P_{JJ_{k|k-1}} - P_{Jz_{k|k-1}} P^{-1}_{zz_{k|k-1}} P^{\top}_{Jz_{k|k-1}}|}\right). \tag{6}$$

B. Sensor Reconfiguration Cost

Sensor reconfiguration cost refers to the cost of moving sensors from one location to another in the grid space. We consider a sensor reconfiguration cost based on the Euclidean distance between the new and previous sensor locations, as illustrated in Fig. 2. Here, d_i^j is the Euclidean distance between the i^{th} grid point and the location of the j^{th} sensor.

Informally, at the ℓ^{th} iteration of the CSCP algorithm, the objective is to find new sensor locations $\boldsymbol{q}^*_{\ell+1}$ that maximize the CRMI while minimizing the cost of sensor reconfiguration from the current configuration $\boldsymbol{q}^*_\ell = \{q_1^{\ell *}, \ldots, q_{N_s}^{\ell *}\}$. To avoid a computationally challenging min-max problem, we modify the CRMI to find the next sensor configuration as

$$\boldsymbol{q}^*_{\ell+1} = \max_{\boldsymbol{q}} \left\{ I^{\text{mod}}(\boldsymbol{q}) := I(J_\ell; z_\ell(\boldsymbol{q})) + \alpha_1 - \alpha_2 \min_{j\in[N_s]} \|q - q_j^{\ell *}\| \right\}, \tag{7}$$

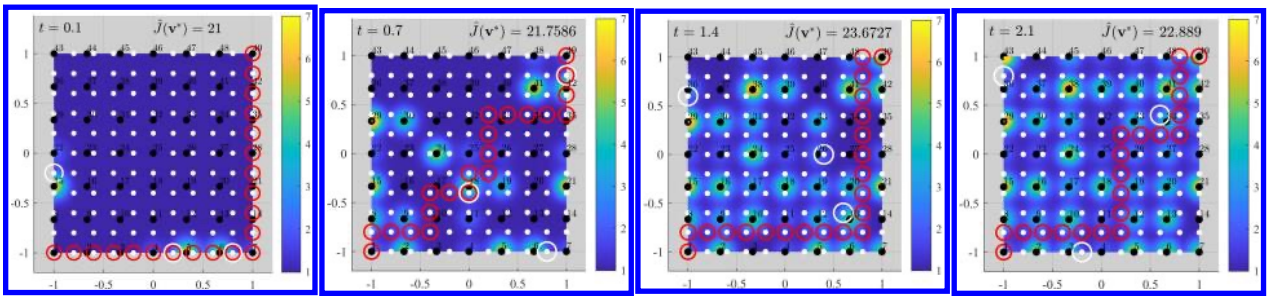
where α_1, α_2 are constants. We choose α_1 to be a relatively large value, e.g., proportional to the size of the overall workspace, whereas α_2 is chosen based on the user's preference for reducing the reconfiguration cost.

C. CSCP Algorithm

Algorithm 1 provides a detailed description of the proposed coupled sensing and planning method. The algorithm sets the initial values to $\widehat{\boldsymbol{\Theta}}_0 = \mathbf{0}$ and $P_0 = \chi \boldsymbol{I}_{(N_P)}$, with χ being a large arbitrary number. It is important to set the parameter estimate prior mean $\widehat{\boldsymbol{\Theta}}_0$ to zero. Due to this "optimistic" prior, the path-planning assumes zero mean threat exposure in areas where sensors have not been placed, and tends to find paths through these areas. In turn, this leads to sensor placement in those unexplored areas.

Intially, an arbitary sensor configuration is chosen, and an optimal path $\boldsymbol{v}_0^*$ of minimum expected cost $\mathbb{E}\left[J(\boldsymbol{v}_0^*)\right]$ is determined. Here, we consider the Dijkstra's algorithm for finding the optimal path $\boldsymbol{v}_k^*$, however the choice of the path planning method is user dependent. At each iteration k, the algorithm computes the path cost variance $\text{Var}[J(\boldsymbol{v}_k^*)]$ of the path $\boldsymbol{v}_k^*$. The algorithm terminates once the variance of the path cost falls below a specified threshold $\epsilon > 0$.

We employ the greedy optimization, in which the sensors are placed sequentially such that the choice of next sensor maximizes the metric $I^{\text{mod}}(\boldsymbol{q})$. For this, we initialize the empty configuration $\boldsymbol{q}_{\text{gr}} = \varnothing$ and perform iterations until N_s sensors are placed. For each sensor in sequence, the CRMI per (6) and the sensor reconfiguration cost at all possible locations $\overline{\boldsymbol{q}}$ are calculated. We implement a greedy approach to account for the sensor reconfiguration cost, considering the minimum travel distance among each of the N_s sensors from their current location to a new location of interest.



(a) $k = 1$ (b) $k = 7$ (c) $k = 14$ (d) $k = 21$

Fig. 3 Visualization of CSCP process for $N_P = 49$, $N_g = 121$, and $N_s = 3$.

The modified metric $I^{\text{mod}}(\boldsymbol{q})$ as per (7) is calculated for all possible locations $\overline{\boldsymbol{q}}$. At each iteration, the greedy optimal configuration is the scalar q^* that maximizes $I^{\text{mod}}(\boldsymbol{q})$. The configuration $\boldsymbol{q}_{\text{gr}}$ is then updated to include q^*.

Once the sensors are optimally placed, new measurements are taken, which are then used to update the state estimate $\widehat{\boldsymbol{\Theta}}$. This iterative process continues until the variance of the estimated path cost $\text{Var}[(\widehat{J}(\boldsymbol{v})]$ falls below the predefined threshold ϵ. Since we implement the greedy optimization of the sensors, the proposed CSCP method has a computational complexity of $O(N_s)$.

Algorithm 1: CSCP Algorithm

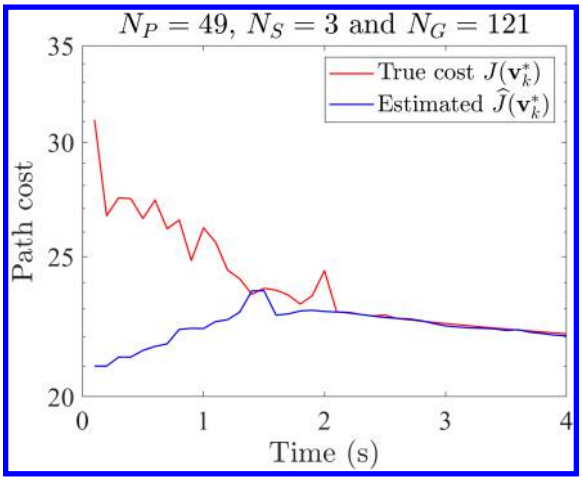
Set $k = 0$, $\widehat{\boldsymbol{\Theta}}_0 = \boldsymbol{0}$, and $P_0 = \chi \boldsymbol{I}_{(N_P)}$;
Initialize sensor placement $\boldsymbol{q}_0 \subset [N_g]$;
Find $\boldsymbol{v}_0^* = \arg\min(\widehat{J}_0(\boldsymbol{v}))$;
while $Var[(J(\boldsymbol{v}_k^*)] > \epsilon$ **do**
  Set $\boldsymbol{q}_{\text{gr}} = \phi$;
  for $j = 1$ *to* N_s **do**
    Set $\overline{\boldsymbol{q}} := [N_g] \backslash \boldsymbol{q}_{\text{gr}}$;
    Calculate $I(J_k; \boldsymbol{z}_k(\overline{\boldsymbol{q}}))$ per (6) and $I^{\text{mod}}(\overline{\boldsymbol{q}})$ per (7);
    Determine $q^* := \arg\max_{q \in \overline{\boldsymbol{q}}}(I^{\text{mod}}(\overline{\boldsymbol{q}}))$;
    $\boldsymbol{q}_{\text{gr}} = \boldsymbol{q}_{\text{gr}} \cup q^*$;
  end
  Obtain new sensor measurements $\boldsymbol{z}_k(\boldsymbol{q}_{\text{gr}})$;
  Update $\widehat{\boldsymbol{\Theta}}_k, P_k$;
  Find $\boldsymbol{v}_k^* := \arg\min(\widehat{J}_k(\boldsymbol{v}))$;
  Increment iteration counter $k = k + 1$;
end

IV. Results and Discussion

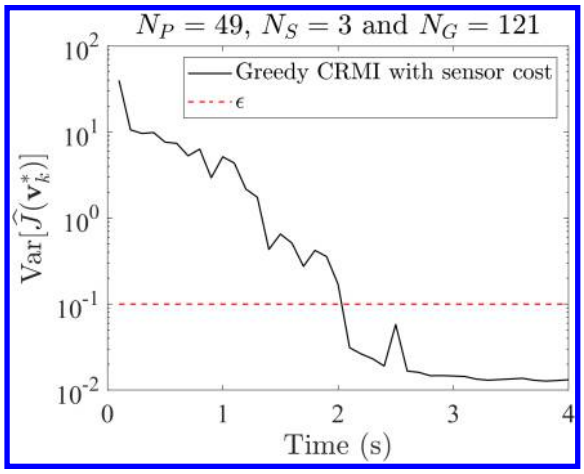
In this section, we first present an illustrative example of the proposed CSCP method. Additionally, we conduct a comparative study between the CSCP method with and without the inclusion of sensor reconfiguration cost. All numerical simulations are conducted within a square workspace $\mathcal{W} = [-1, 1] \times [-1, 1]$ using non-dimensional units.

A. Illustrative Example

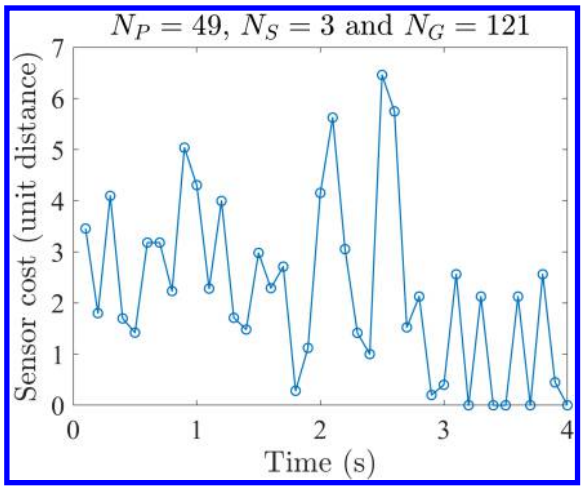
Figure 3 shows the visualization of the CSCP method implemented on an illustrative example across different time steps. The number of threat parameters, sensors, and the grid points used for the analysis are $N_P = 49$, $N_s = 3$, and



(a) Path cost mean.



(b) Path cost variance.

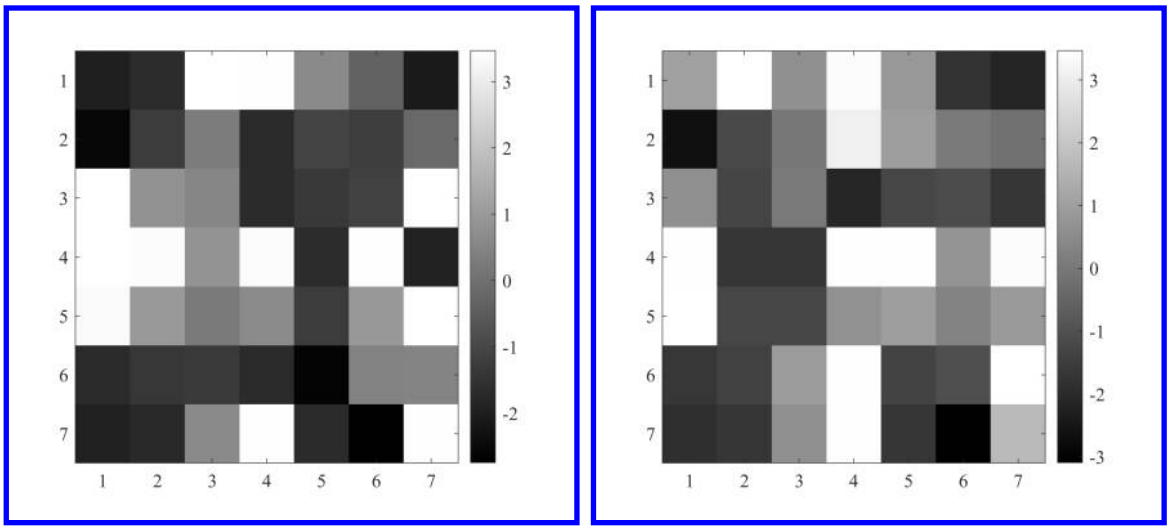


(c) Sum of distances traveled by sensors.

Fig. 4 Evolution of path cost mean and variance over CSCP iterations.

$N_g = 121$, respectively. The initial and goal points are represented by the grid points at the bottom-left and top-right corners of the map, respectively. The white dots in the Fig. 3 represent the grid points, while the threat parameters, represented by the black dots are uniformly spaced within the workspace. The red circles indicate the path $\boldsymbol{v}_k^*$ with the minimum estimated cost, while the white circles represent the location of sensors. For $k = 1$, as shown in Fig. 3(a), there is uncertainty in the environment, thus the obtained path is the shortest path between the initial and goal points. As the CSCP process proceeds, the threat parameter estimation uncertainty is reduced. Figures 3(b) and (c) show the evolution of $\widehat{c}$, optimal sensor placement and the local optimal path obtained for intermediate time steps, $k = 7$ and 11. At $k = 21$ iterations, the specified threshold criteria $\mathrm{Var}[J(\boldsymbol{v}^*)] \leqslant 0.1$ is satisfied, and the obtained path represents the optimal path.

Figure 4(a) shows the comparison between the true and estimated mean path cost. Initially, the estimated mean path cost is much lower than the true path cost. This is because there is not much information about the environment, and the estimator relies heavily on the "optimistic" prior, which results in threat estimates with small values. Upon termination at $k = 21$, the true path cost $J(\boldsymbol{v}_k^*) = 22.91$ closely aligns with the estimated path cost $\widehat{J}(\boldsymbol{v}_k^*) = 22.89$. The convergence of the CSCP method is shown by a path cost variance plot in Fig. 4(b). As the iterative process continues, the path cost variance $\mathrm{Var}[(\widehat{J}(\boldsymbol{v}_k^*)]$ decreases, and the algorithm terminates when $\mathrm{Var}[(\widehat{J}(\boldsymbol{v}_k^*)]$ falls below $\epsilon = 0.1$. Figure 4(c) shows the sensor reconfiguration cost values at different time steps. For the computation of sensor reconfiguration cost, we choose $\alpha_1 = \sqrt{8}$ (diagonal distance across the workspace) and $\alpha_2 = 0.01$. For $N_s = 3$, the sensor cost is the cumulative



(a) (b)

Fig. 5 log **diagonal values of error covariance at the final iteration of the greedy CSCP with sensor reconfiguration cost (a) and without sensor reconfiguration cost (b).**

sum of the distance traveled by the three sensors at each iterations.

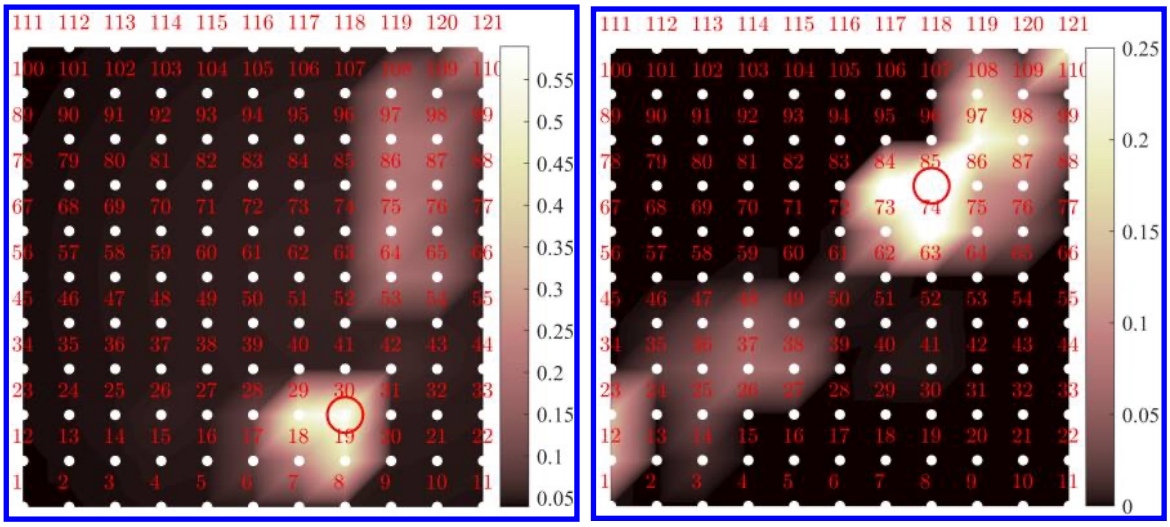
Figure 5(a) shows the estimation error covariance P at the final iteration, mapped to the spatial regions of the environment using the centers of spatial basis functions Φ. In a slight departure from convention, Fig. 5 shows the *logarithms* of the diagonal values of P, which explains the negative values despite P being symmetric positive definite. The white regions are those with high estimation error covariance, where no or few sensors are placed throughout the execution of the CSCP process. The black regions indicate areas where sensors are frequently placed, resulting in low uncertainty within those regions.

B. Comparative Study

We performed a comparison by implementing the greedy CSCP algorithm without including the sensor reconfiguration cost in the same example as discussed earlier. Figures 5(a) and (b) show the estimation error covariance P (diagonal values) at the final CSCP iteration with and without sensor reconfiguration cost. In the earlier case, the sensor placement is performed such that the regions with low threat estimation error are close to optimal path found in Fig. 3(d), which is not the case when the sensor reconfiguration cost is not considered.

Figure 6 shows the intensity map of the mutual information values at each grid point for the illustrative example. Note that the CRMI $I(J; z(q))$ and the modified metric $I^{\text{mod}}(z(q))$ values are computed for $k = 21$ with a single sensor. In Fig. 6(a), the sensor is placed at the grid point numbering 30, which has the maximum CRMI value. Figure 6(b) shows the intensity map of the CRMI metric, in which a sensor is placed at grid point 85. It can be observed that the $I(J; z(q))$ values are concentrated in a narrow region that would resemble the vicinity of the path rather than uniformly distributed within the environment.

A comparative example of sensor placements for $N_P = 49$, $N_g = 121$, and $N_s = 4$ using CSCP with and without consideration of the sensor configuration cost is shown in Fig. 7(a) and (b). Red circles in the figure indicate sensor positions based on maximizing the CRMI metric, whereas the black circles indicate sensor locations based on maximizing the modified metric. At $k = 20$, it can be observed that the two sensor locations in the grid space are common for methods. As expected, when sensor reconfiguration cost is considered, the new sensor locations (black circles) at $k = 21$ are relatively closer to the previous locations at $k = 20$ as compared to sensor positions denoted by the red circles across the two iterations. Figure 7(c) shows the values of the CRMI and the modified CRMI for different numbers of sensors.



(a) Modified metric with sensor cost $I^{\text{mod}}(J;z(q))$. (b) CRMI without sensor cost $I(J;z(q))$.

Fig. 6 CRMI and modified CRMI at the final iteration of the illustrative example.

V. Conclusions

In this paper, we discussed an iterative method for simultaneous and coupled sensor placement and path-planning for a mobile agent to navigate in a threat field. Optimal sensor locations are identified by maximizing a metric that collectively takes into account a context-relevant mutual information (CRMI) metric and the sensor reconfiguration cost. The CRMI addresses uncertainty reduction in the path cost, while the sensor reconfiguration cost addresses distance traveled by the sensor during reconfiguration. The proposed CSCP algorithm iteratively places the sensors at an optimal set of location, updates the environment threat estimate, and plans path with minimum expected cost. We conducted numerical situations with an example to show the visualization of the CSCP process. A comparative study between the CSCP method with and without sensor reconfiguration cost is carried out.

Acknowledgments

This work is funded in part by the US NSF-DCSD-CMMI grant #2126818.

References

[1] Valavanis, K. P., and Vachtsevanos, G. J., *Handbook of unmanned aerial vehicles*, Vol. 1, Springer, 2015.

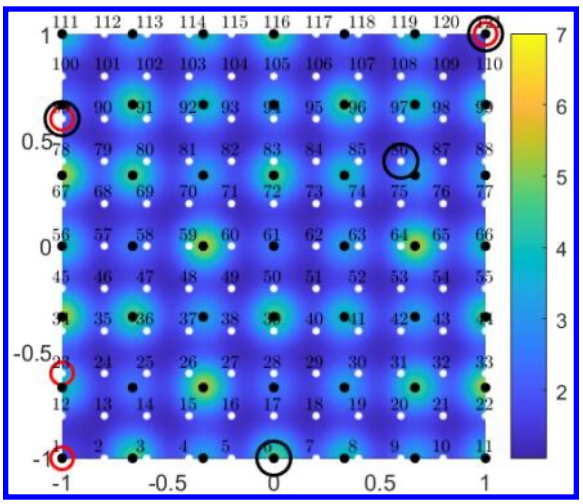
[2] Crassidis, J. L., and Junkins, J. L., *Optimal estimation of dynamic systems*, CRC Press, Talyor & Francis Group, Boca Raton, FL, USA, 2012.

[3] Catlin, D. E., *Estimation, control, and the discrete Kalman filter*, Vol. 71, Springer Science & Business Media, 2012.

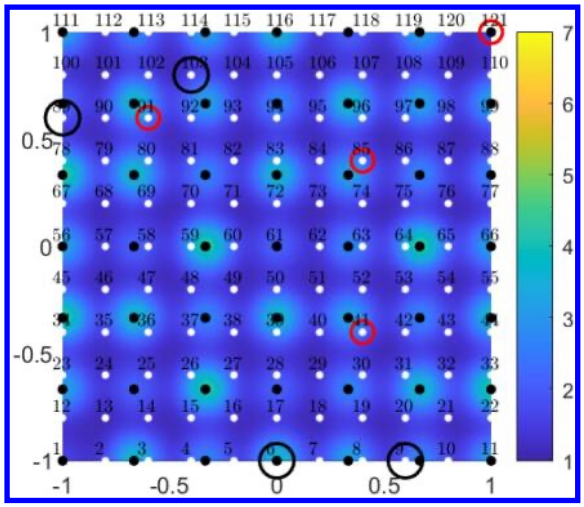
[4] Lewis, F. L., Xie, L., and Popa, D., *Optimal and robust estimation: with an introduction to stochastic control theory*, CRC press, 2017.

[5] Julier, S. J., and Uhlmann, J. K., "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, Vol. 92, 2004, pp. 401–422. doi:10.1109/JPROC.2003.823141.

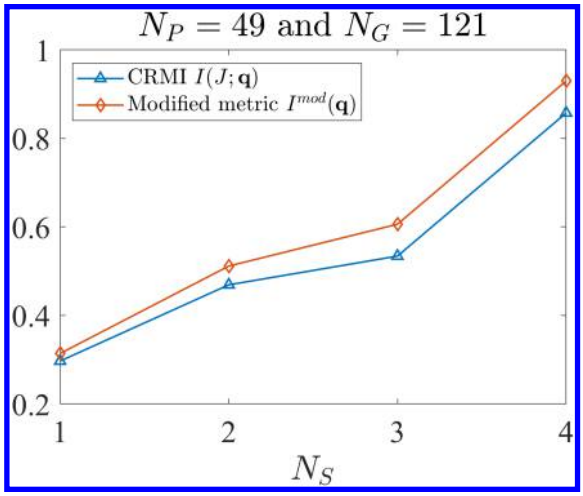
[6] Evensen, G., "The ensemble Kalman filter: Theoretical formulation and practical implementation," *Ocean dynamics*, Vol. 53, 2003, pp. 343–367.



(a) Sensor locations at $k = 20$.



(b) Sensor locations at $k = 21$.



(c) CRMI and modified CRMI.

Fig. 7 Comparison between the CSCP method with and without sensor reconfiguration cost.

[7] Djuric, P. M., Kotecha, J. H., Zhang, J., Huang, Y., Ghirmai, T., Bugallo, M. F., and Miguez, J., "Particle filtering," *IEEE signal processing magazine*, Vol. 20, No. 5, 2003, pp. 19–38.

[8] Thrun, S., Burgard, W., and Fox, D., *Probabilistic Robotics*, The MIT Press, 2006.

[9] Lavalle, S. M., *Planning Algorithms*, Cambridge University Press, 2006. URL http://planning.cs.uiuc.edu/.

[10] Patle, B. K., L, G. B., Pandey, A., Parhi, D. R., and Jagadeesh, A., "A review: On path planning strategies for navigation of mobile robot," *Defence Technology*, Vol. 15, 2019, pp. 582–606. doi:10.1016/j.dt.2019.04.011.

[11] Rückin, J., Jin, L., and Popović, M., "Adaptive Informative Path Planning Using Deep Reinforcement Learning for UAV-based Active Sensing," *International Conference on Robotics and Automation (ICRA)*, 2022, pp. 4473–4479. doi:10.1109/ICRA46639.2022.9812025.

[12] Wen, T., Wang, X., Zheng, Z., and Sun, Z., "A DRL-based path planning method for wheeled mobile robots in unknown environments," *Computers and Electrical Engineering*, Vol. 118, 2024, p. 109425.

[13] Qin, Y., Zhang, Z., Li, X., Huangfu, W., and Zhang, H., "Deep reinforcement learning based resource allocation and trajectory planning in integrated sensing and communications UAV network," *IEEE Transactions on Wireless Communications*, Vol. 22, No. 11, 2023, pp. 8158–8169.

[14] Kamil, F., and Moghrabiah, M. Y., "Multilayer decision-based fuzzy logic model to navigate mobile robot in unknown dynamic environments," *Fuzzy Information and Engineering*, Vol. 14, No. 1, 2022, pp. 51–73.

[15] Popović, M., Ott, J., Rückin, J., and Kochenderfer, M. J., "Learning-based methods for adaptive informative path planning," *Robotics and Autonomous Systems*, 2024, p. 104727.

[16] Chen, J., Du, C., Zhang, Y., Han, P., and Wei, W., "A clustering-based coverage path planning method for autonomous heterogeneous UAVs," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 12, 2021, pp. 25546–25556.

[17] Chintam, P., Lei, T., Osmanoglu, B., Wang, Y., and Luo, C., "Informed sampling space driven robot informative path planning," *Robotics and Autonomous Systems*, Vol. 175, 2024, p. 104656.

[18] Kangsheng, T., and Guangxi, Z., "Sensor management based on fisher information gain," *Journal of Systems Engineering and Electronics*, Vol. 17, 2006, pp. 531–534.

[19] Takahashi, S., Sasaki, Y., Nagata, T., Yamada, K., Nakai, K., Saito, Y., and Nonomura, T., "Sensor Selection by Greedy Method for Linear Dynamical Systems: Comparative Study on Fisher-Information-Matrix, Observability-Gramian and Kalman-Filter-Based Indices," *IEEE Access*, Vol. 11, 2023, pp. 67850–67864. doi:10.1109/ACCESS.2023.3291415.

[20] Wang, H., Yao, K., Pottie, G., and Estrin, D., "Entropy-based Sensor Selection Heuristic for Target Localization," *3rd Int. Symp. Information Processing in Sensor Networks*, 2004, pp. 36–45. doi:10.1145/984622.984628.

[21] Blasch, E. P., Maupin, P., and Jousselme, A.-L., "Sensor-based allocation for path planning and area coverage using UGSs," *Proceedings of the IEEE 2010 National Aerospace & Electronics Conference*, 2010, pp. 361–368. doi:10.1109/NAECON.2010.5712978.

[22] Krause, A., Singh, A., and Guestrin, C., "Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies," *Journal of Machine Learning Research*, Vol. 9, 2008, pp. 235–284.

[23] Schmidt, K., Smith, R. C., Hite, J., Mattingly, J., Azmy, Y., Rajan, D., and Goldhahn, R., "Sequential optimal positioning of mobile sensors using mutual information," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, Vol. 12, No. 6, 2019, pp. 465–478.

[24] Adurthi, N., Singla, P., and Majji, M., "Mutual information based sensor tasking with applications to space situational awareness," *Journal of Guidance, Control, and Dynamics*, Vol. 43, 2020, pp. 767–789. doi:10.2514/1.G004399.

[25] Ranieri, J., Chebira, A., and Vetterli, M., "Near-optimal sensor placement for linear inverse problems," *IEEE Transactions on Signal Processing*, Vol. 62, 2014, pp. 1135–1146. doi:10.1109/TSP.2014.2299518.

[26] Ott, J., Kochenderfer, M. J., and Boyd, S., "Approximate Sequential Optimization for Informative Path Planning," *arXiv preprint arXiv:2402.08841*, 2024.

[27] Soderlund, A. A., and Kumar, M., "Optimization of multitarget tracking within a sensor network via information-guided clustering," *Journal of Guidance, Control, and Dynamics*, Vol. 42, 2019, pp. 317–334. doi:10.2514/1.G003656.

[28] Robbiano, C., Azimi-Sadjadi, M. R., and Chong, E. K., "Information-Theoretic Interactive Sensing and Inference for Autonomous Systems," *IEEE Transactions on Signal Processing*, Vol. 69, 2021, pp. 5627–5637.

[29] Hoffmann, F., Charlish, A., Ritchie, M., and Griffiths, H., "Sensor Path Planning Using Reinforcement Learning," *IEEE 23rd International Conference on Information Fusion (FUSION)*, 2020, pp. 1–8. doi:10.23919/FUSION45008.2020.9190242.

[30] Liu, X., Phan, D., Hwang, Y., Klein, L., Liu, X., and Yeo, K., "Optimal Sensor Allocation with Multiple Linear Dispersion Processes," *arXiv preprint arXiv:2401.10437*, 2024.

[31] Seth, G., Bhushan, M., and Patwardhan, S. C., "Optimal Sensor Placement Design for Profile Estimation of Distributed Parameter Systems," *Industrial & Engineering Chemistry Research*, 2024.

[32] Manohar, K., Brunton, B. W., Kutz, J. N., and Brunton, S. L., "Data-driven sparse sensor placement for reconstruction: Demonstrating the benefits of exploiting known patterns," *IEEE Control Systems Magazine*, Vol. 38, No. 3, 2018, pp. 63–86.

[33] Hussain, L. A., Singh, S., Mizouni, R., Otrok, H., and Damiani, E., "A predictive target tracking framework for IoT using CNN–LSTM," *Internet of Things*, Vol. 22, 2023, p. 100744.

[34] Inoue, T., Ikami, T., Egami, Y., Nagai, H., Naganuma, Y., Kimura, K., and Matsuda, Y., "Data-driven optimal sensor placement for high-dimensional system using annealing machine," *Mechanical Systems and Signal Processing*, Vol. 188, 2023, p. 109957.

[35] Wang, Z., Li, H. X., and Chen, C., "Reinforcement learning-based optimal sensor placement for spatiotemporal modeling," *IEEE Transactions on Cybernetics*, Vol. 50, 2020, pp. 2861–2871. doi:10.1109/TCYB.2019.2901897.

[36] Cooper, B. S., and Cowlagi, R. V., "Interactive planning and sensing in unknown static environments with task-driven sensor placement," *Automatica*, Vol. 105, 2019, pp. 391–398. doi:10.1016/j.automatica.2019.04.014.

[37] Laurent, C. S., and Cowlagi, R. V., "Near-optimal task-driven sensor network configuration," *Automatica*, Vol. 152, 2023. doi:10.1016/j.automatica.2023.110966.

[38] Fang, J., Zhang, H., and Cowlagi, R. V., "Interactive Route-Planning and Mobile Sensing with a Team of Robotic Vehicles in an Unknown Environment," *AIAA Scitech 2021 Forum*, 2021, p. 0865.

[39] Poudel, P., and Cowlagi, R. V., "Coupled Sensor Configuration and Planning in Unknown Dynamic Environments with Context-Relevant Mutual Information-based Sensor Placement," *2024 American Control Conference (ACC)*, IEEE, 2024, pp. 306–311.

[40] Leong, A. S., Quevedo, D. E., Ahlén, A., and Johansson, K. H., "Network topology reconfiguration for state estimation over sensor networks with correlated packet drops," *IFAC Proceedings Volumes*, Vol. 47, No. 3, 2014, pp. 5532–5537.

[41] Ramachandran, G. S., Daniels, W., Matthys, N., Huygens, C., Michiels, S., Joosen, W., Meneghello, J., Lee, K., Canete, E., Rodriguez, M. D., et al., "Measuring and modeling the energy cost of reconfiguration in sensor networks," *IEEE Sensors Journal*, Vol. 15, No. 6, 2015, pp. 3381–3389.

[42] Grichi, H., Mosbahi, O., Khalgui, M., and Li, Z., "New power-oriented methodology for dynamic resizing and mobility of reconfigurable wireless sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 48, No. 7, 2017, pp. 1120–1130.