
FA9550-23-1-0756

**Real-time optimal distributed estimation and control of
spatiotemporal processes using multi-domain methods and optimally-
guided mobile sensors**

Michael A. Demetriou and Nikolaos A. Gatsonis

Aerospace Engineering Department

Worcester Polytechnic Institute

Worcester, MA, 01609

and

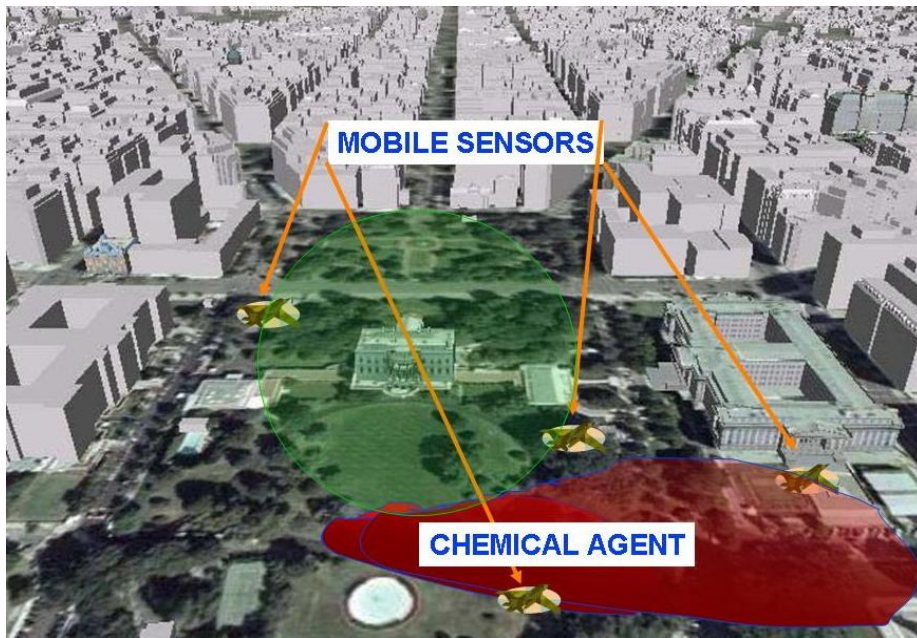
Weiwei Hu (join task with FA9550-23-1-0675)

AFOSR Dynamical Systems and Control Theory Program Review

Friday August 30, 2024

Main idea: adaptive sampling in spatially distributed systems

- Consider the following
 - Chemical substance released by stationary or mobile ground vehicle in city block
 - Chemical/biological substance released by flying agent (adversary)



Motivation

- Address sensor technology limitations in providing spatial gradients of measured quantities
 - Sensors cannot measure spatial gradients for mobile sensor guidance
- Address real-time CFD of PDEs over large spatial domains
 - Cannot solve Riccati equations over large domains ($20\text{km} \times 5\text{ km} \times 2\text{ km}$)
 - Must pay attention to spatial resolution
 - Use (computational) domain decomposition \rightarrow parallel processing
- Must develop theories (well-posedness, filter kernel sparsity, sensor kernel classification, optimization) to address technology limitations
 - Use physical and computational domain decompositions
 - Solve filter Riccati equations over smaller spatial domains (physical domain decomposition)
 - Enforce/guarantee sparsity of filter kernel

Project objectives

Goals

- Develop an advanced theoretical and computational approach that enables the *real-time estimation* of spatiotemporal processes modelled by PDEs using mobile sensors

Objectives

- Develop an abstract framework for the estimator design of PDEs using DD methods and mobile sensors.
- Develop sensor type classifications, represented by the sensor spatial distributions, producing filter kernels that vanish outside the inner domain and thus produce (i) an optimal Kalman filter in the inner domain and (ii) a naive observer in the outer domain.
- Develop a guidance scheme for the mobile sensor that optimizes the hybrid estimator performance.
- Develop and implement the numerical scheme hybrid estimator using a DD method with finite volume evaluation of the PDE terms and a multi-step integration scheme with sub-cycling.

Intellectual merit

- Plant equations
 - Represented by 3D advection diffusion PDEs in very large spatial domains (e.g. city block)
- Computational model
 - Use OpenMP heterogeneous, nonoverlapping domain-decomposition, explicit, finite volume method (HT-NODDE-FVM) with Total Variation Diminishing (TVD)-RK for Hybrid Estimator Grid Generation and OpenMP
 - Use computational domain decomposition for parallel processing
 - Enables truly **real-time** estimation using sensor measurements
- State estimation
 - Implement hybrid domain decomposition filters
 - Physical domain decomposition partitions spatial domain into two domains with Kalman/Luenberger observer in inner domain and naïve observer in outer domain
 - Use sensor(s) onboard UAVs
 - Sensor guidance dictated by estimator performance

How to view things

The plant is the PDE that describes the plume dispersion in space and time, gives rise to state operator (A matrix)

- As opposed to the traditional approach in Zermelo's navigation problem in which the plant were the motion equations and the river currents were the disturbance.

The moving source (intruder) is the exosystem

- Intruder vehicle obeys kinematic and dynamic equations of a mobile platform (a/c here)
- The location and motion of the intruder vehicle is the input operator (B matrix)
- The input signal of the intruder to the plant is the mass release rate

The measurements are obtained by a moving sensor

- Concentration sensor onboard a mobile platform
- Location of sensor (barycenter of mobile platform) described by output operator (C matrix)
- Mobile platform obeys kinematic and dynamic equations (not a point mass)

Control & estimation objective: move the sensor to improve estimator

PDE Estimation

Process modeled by the 3D advection-diffusion PDE with in-domain actuation and Dirichlet B.C.'s

$$\begin{aligned}\partial_t x(t, \xi) &= \alpha \Delta x(t, \xi) - \beta(\xi) \cdot \nabla x(t, \xi) - \gamma(\xi) x(t, \xi) \\ &\quad + b(\xi) f(t) \quad \text{in } \Omega \times (0, \infty) \\ x(t, \xi) &= 0, \quad \text{on } \partial\Omega \times (0, \infty) \\ x(t, \xi) &= x_0(\xi), \quad \text{on } \Omega \times \{t = 0\}\end{aligned}$$

$$y(t) = \int_{\Omega} c(\xi) x(t, \xi) d\xi \quad \text{in } (0, \infty)$$

- $x(t, \xi)$ the state at time t and spatial coordinates $\xi = (\xi_1, \xi_2, \xi_3) \in \Omega \subset \mathbb{R}^3$
- $b(\xi)$ the spatial distribution of actuating devices and/or sources
- $y(t)$ is the sensor measurement, $c(\xi)$ is the sensor distribution

Assume $\beta \in (W^{1,\infty}(\Omega))^d$ and $\gamma \in L^\infty(\Omega)$ with positive diffusivity $\alpha > 0$ and square integrability of forcing term for well-posedness

PDE filter over contiguous domain

estimator is a copy of the plant plus **output injection** (innovation)

$$\begin{aligned}\partial_t \hat{x}(t, \xi) &= \alpha \Delta \hat{x}(t, \xi) - \beta(\xi) \cdot \nabla \hat{x}(t, \xi) - \gamma(\xi) \hat{x}(t, \xi) \\ &\quad + b(\xi) f(t) + \lambda^T(\xi) \left(y(t) - \int_{\Omega} c(\xi) \hat{x}(t, \xi) d\xi \right) \text{ in } \Omega \times (0, \infty) \\ \hat{x}(t, \xi) &= 0, & \text{on } \partial\Omega \times (0, \infty) \\ \hat{x}(t, \xi) &= \hat{x}_0(\xi) \neq x_0(\xi), & \text{on } \Omega \times \{t = 0\}\end{aligned}$$

- $\hat{x}(t, \xi)$ is the estimate of the plant state $x(t, \xi)$
- $\lambda(\xi)$ is the filter kernel
 - can be designed using Luenberger observer or Kalman filter
- $\lambda(\xi)$ related to the adjoint of the filter operator L

$$\begin{aligned}y^T(t) \int_{\Omega} \lambda(\xi) \phi(\xi) d\xi &= (y(t), L^* \phi) \\ &= (Ly(t), \phi), \quad \phi \in H_0^1(\Omega)\end{aligned}$$

Plant in abstract form

consider plant and estimator in weak form:

- L^2 inner product for interior of Ω denoted by (\cdot, \cdot)
- L^2 inner product at the boundary $\partial\Omega$ denoted by $\langle \cdot, \cdot \rangle$
- Interpolating spaces: Gelfand triple $V \hookrightarrow H \hookrightarrow V^*$
 - State space $H = L^2(\Omega)$ is the pivot space
 - $V = H_0^1(\Omega)$ and $V^* = H^{-1}(\Omega)$ is the conjugate dual

plant in strong form (use notation $x(t) = x(t, \cdot)$)

$$\begin{aligned}\dot{x}(t) &= Ax(t) + B_1 u(t) + B_2 g(t) \quad \text{in } V^* \\ x(0) &= x_0 \\ y(t) &= Cx(t)\end{aligned}$$

$Bf(t) = B_1 u(t) + B_2 g(t)$, $u(t)$ is controls, $g(t)$ is disturbance

Filter in abstract form

The output operator is

$$C\phi = \int_{\Omega} c(\xi) \phi(\xi) d\xi, \quad \phi \in V$$

Estimator in abstract form (Luenberger or Kalman)

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + B_1 u(t) + L(y(t) - C\hat{x}(t)) \quad \text{in } V^* \\ \hat{x}(0) &= \hat{x}_0 \neq x_0 \end{aligned}$$

Estimation error $e(t) = x(t) - \hat{x}(t)$

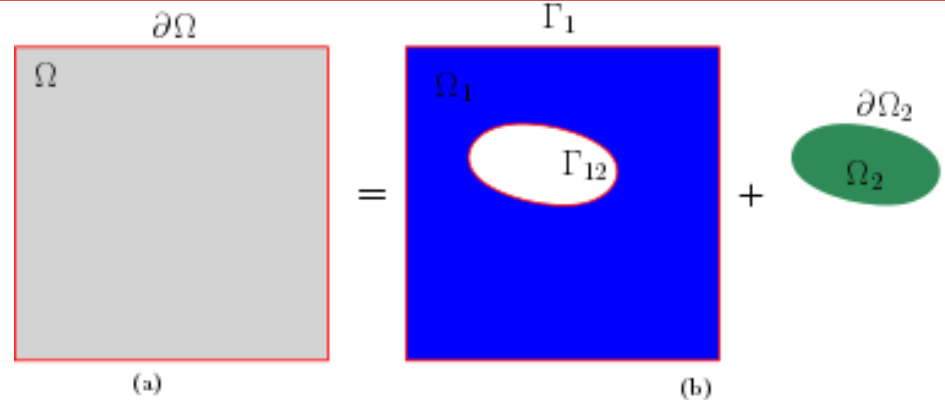
$$\begin{aligned} \dot{e}(t) &= Ae(t) + B_2 g(t) - L(y(t) - C\hat{x}(t)) \quad \text{in } V^* \\ e(0) &= x_0 - \hat{x}_0 \neq 0 \end{aligned}$$

Filter gain (filter kernel) selection

- Luenberger design: select L s.t. $(A - LC)$ generates exp. stable semigroup, simple choice $L = kC^*$
- Kalman filter: solve operator equation for $L = \Sigma C^* R^{-1}$
$$\Sigma A^* + A\Sigma - \Sigma C^* R^{-1} C \Sigma + Q = 0 \quad \text{in } V^*$$

Domain decomposition filter

state estimator is considered in the decomposed domain $\Omega = \Omega_1 \cup \Omega_2$ depicted in the figure



Can write the *domain decomposition filter* over each of the subdomains; define the spaces $V_1 = \{\phi \in H^1(\Omega_1): \phi = 0 \text{ on } \Gamma_1\}$ and $V_2 = H^1(\Omega_2)$ with $\Gamma_1 = \partial\Omega_1 \cap \partial\Omega$ the outer domain on which the Dirichlet boundary condition is imposed on the boundary

Define $v_1 = x|_{\Omega_1}$ and $v_2 = x|_{\Omega_2}$ giving the transmission conditions where η_1 and η_2 denote the unit outward normal vectors w.r.t. Ω_1 and Ω_2

$$\begin{aligned} v_1 &= v_2 && \text{on } \Gamma_{12} \times (0, \infty) \\ \frac{\partial v_1}{\partial \eta_1} &= -\frac{\partial v_2}{\partial \eta_2} && \text{on } \Gamma_{12} \times (0, \infty) \end{aligned}$$

DD filter

Estimator over the two subdomains, with $\hat{x}_1(t) = \hat{v}_1(t, \cdot)$ and $\hat{x}_2(t) = \hat{v}_2(t, \cdot)$, is abstractly given by

$$\begin{aligned}\dot{\hat{x}}_1(t) &= A\hat{x}_1(t) + D_1^* N_1 L_{D_1} D_2 \hat{x}_2(t) + L_1(y(t) - C\hat{x}_1(t)) \quad \text{in } V_1^* \\ \dot{\hat{x}}_2(t) &= A\hat{x}_2(t) + D_2^* N_2 L_{D_2} D_1 \hat{x}_1(t) + L_2(y(t) - C\hat{x}_2(t)) \quad \text{in } V_2^*\end{aligned}$$

where D_i and N_i , $i = 1, 2$, are the Dirichlet and Neumann trace operators $D_i: \Omega_i \rightarrow \Gamma_{12}$, $D_i v_i = v_i|_{\Gamma_{12}}$ and $N_i: \Omega_i \rightarrow \Gamma_{12}$, $N_i v_i = \frac{\partial v_i}{\partial \eta_i}$ and L_{D_i} , $i = 1, 2$, denote the Dirichlet lifting operators; We have $D_1 v_1 = D_2 v_2$

The filter operator L is computed from the filter for the state estimator defined over the *entire domain* Ω . The individual filter kernels are given by

$$\begin{aligned}(Ly, \phi) &= (y, L^* \phi) = y^T \int_{\Omega} \lambda(\xi) \phi(\xi) d\xi \\ &= y^T \int_{\Omega_1} \lambda_1(\xi) \phi(\xi) d\xi + y^T \int_{\Omega_2} \lambda_2(\xi) \phi(\xi) d\xi\end{aligned}$$

Hybrid DD filter

The filter operator L is calculated from the filter design (Kalman or Luenberger) for the single domain system and is appropriately projected in the two subdomains in order to simulate the 2 estimator states $\hat{x}_1(t)$, $\hat{x}_2(t)$

- Requires the solution to the operator Riccati equation (if Kalman filter design is used) over the **entire domain** Ω
- In its realization, one must solve a matrix Riccati equation associated with the finite dimensional representation of the infinite dimensional system
- Requires the solution to a high dimensional matrix Riccati equation!

If instead, one solves a separate Riccati equation over the inner domain Ω_2 , the resulting filter gain is computed from a *much smaller dimensional* matrix Riccati equation

Hybrid DD filter

Hybrid DD estimator: implements estimator only over inner domain Ω_2

$$\begin{aligned}\dot{\hat{x}}_1(t) &= A\hat{x}_1(t) + D_1^*N_1L_{D_1}D_2\hat{x}_2(t) && \text{in } V_1^* \\ \dot{\hat{x}}_2(t) &= A\hat{x}_2(t) + D_2^*N_2L_{D_2}D_1\hat{x}_1(t) + \bar{L}_2(y(t) - C\hat{x}_2(t)) && \text{in } V_2^*\end{aligned}$$

$\bar{L}_2 = \Sigma_2 C^* R^{-1}$ is the hybrid filter operator given via the solution to the modified operator filter Riccati equation

$$\boxed{\Sigma_2: \quad \dot{\Sigma}_2 = A\Sigma_2 + \Sigma_2 A^* - \Sigma_2 C^* R^{-1} C \Sigma_2 + Q \quad \text{in } V_2^*}$$

- implementation of the *hybrid* DD estimator requires the solution to a significantly lower dimensional matrix Riccati equation
- associated state estimator in outer domain Ω_1 becomes a naïve observer due to the *transmission conditions*, represented by the term $D_2^*N_2L_{D_2}D_1\hat{x}_1(t)$ which is viewed as a consensus protocol
- outer domain observer $\hat{x}_1(t)$ receives information from the inner domain and becomes a Luenberger observer

Hybrid DD filter

Caveat: The *hybrid* DD filter is not derived by simply selecting the sensor kernel function $c(\xi)$ with $\text{supp}(c(\cdot)) \subseteq \bar{\Omega}_2$. If it were so, then one would have

$$\int_{\Omega_1} c(\xi) \hat{x}_1(t, \xi) d\xi = 0$$

(i.e., $C\hat{x}_1(t) \equiv 0$), and the DD filter

$$\dot{\hat{x}}_1(t) = A\hat{x}_1(t) + D_1^* N_1 L_{D_1} D_2 \hat{x}_2(t) + L_1 (y(t) - C\hat{x}_1(t)) \text{ in } V_1^*$$

would become

$$\dot{\hat{x}}_1(t) = A\hat{x}_1(t) + D_1^* N_1 L_{D_1} D_2 \hat{x}_2(t) + L_1 y(t) \text{ in } V_1^*$$

In this case, output $y(t)$ acts as a disturbance in the outer domain filter!!

Hybrid DD filter

Definition: a DD filter becomes a *hybrid* DD filter if the filter kernel λ is nullified over region Ω_1

For this project: examine conditions under which the operator filter gain L when restricted (projected) in the outer region Ω_1 it is zero, i.e., find conditions under which $L_1 = L|_{\Omega_1} = 0$

When designing a *hybrid* DD filter with the filter kernel based on Kalman filter, one must impose constraints on the estimation problem so that

$$\lambda|_{\Omega_2} = \lambda_2 \quad \text{and} \quad \lambda|_{\Omega_1} = 0$$

Hybrid DD filter

Case 1: output operator is nullified in Ω_1 , i.e., $C_1 = C|_{\Omega_1} = 0$, and filter operator $L = kC^*$ then $L_1 = 0$ naturally, i.e., have a sparse filter kernel

Lemma. There exist a $k^* = k^*(\varepsilon)$ s.t. for $k \geq k^*$

$$((A_0 - kLC)\phi, \phi) \geq (\mu_1(\Omega_1) - \varepsilon)\|\phi\|_{L^2}^2, \quad \forall \phi \in V \left(= H_0^1(\Omega)\right)$$

where $\mu_1(\Omega_1)$ is the first eigenvalue of A_1 and

$$(A_0\phi, \psi) = \alpha \int_{\Omega} \nabla\phi \nabla\psi \, d\xi, \text{ for } \phi, \psi \in V$$

Theorem. For $\varepsilon > 0$ suff. small, choose $k \geq k^*$ such that the c.l. operator positivity is satisfied. Then there exists a unique solution to the error dynamics which satisfies

$$\|e\|_{L^2}^2 \leq \exp\{-2at\}\|e_0\|_{L^2}^2$$

$$\text{here, } a = \left(\mu_1(\Omega_1) - \varepsilon - \left\|\frac{1}{2}\nabla\cdot\beta - \gamma\right\|_{L^\infty}\right)$$

Hybrid DD filter

Theory can also handle moving sensors

Corollary. For $\varepsilon > 0$, if there exist $k^* = k^*(t, \varepsilon)$ s.t. for $k \geq k^*$

$$\left((A - L(t, k)C(t))\phi, \phi \right) \geq (\mu_1(\Omega_1(t)) - \varepsilon) \|\phi\|_{L^2}^2$$

for $\forall \phi \in V$ and $t > 0$ a.e., then there exists a unique solution to the error dynamics with mobile sensors, which satisfies the exponential decay rate, that is,

$$\|e\|_{L^2}^2 \leq \exp\{-2a(t)\} \|e_0\|_{L^2}^2$$

here, $a(t) = \int_0^t \left(\mu_1(\Omega_1(\tau)) - \varepsilon - \left\| \frac{1}{2} \nabla \cdot \beta - \gamma \right\|_{L^\infty} \right) d\tau$

Remark. Can also handle sparsity in time for observations

Remark. Can handle the case of unbounded output operators (point measurements; solution not pointwise measurable in space since V not embedded in $C(\Omega)$), but carefully modify to apply above results.

Hybrid DD filter

Generally, impossible to classify the feedback kernel sparsity for the case of Kalman-based filter gain computation

For this project, we consider two cases:

Case 2a. (restricted Kalman filter) select Q with $L = C^*R^{-1}$ and $C_1 = C|_{\Omega_1} = 0$ so that $\Sigma = I$

$$\begin{aligned} 0 &= \Sigma A^* + A \Sigma - \Sigma C^* R^{-1} C \Sigma + Q \quad \text{in } V^* \\ &= A^* + A - C^* R^{-1} C + Q \end{aligned}$$

i.e., argue that for a specific Q and prescribed $L = C^*R^{-1}$, one has an optimal *hybrid* DD filter

Case 2b. (spatial frequency weighting) select Q so that the solution

$$0 = \Sigma A^* + A \Sigma - \Sigma C^* R^{-1} C \Sigma + Q \quad \text{in } V^*$$

ensures $L_1 = L|_{\Omega_1} = 0$

Hybrid DD filter

Clarifications

The state operator is time invariant, but the DD filter has time varying state operators $A_1(t)$ and $A_2(t)$ because of the motion of subdomain Ω_2

The output operator is time varying because of the motion of the sensor

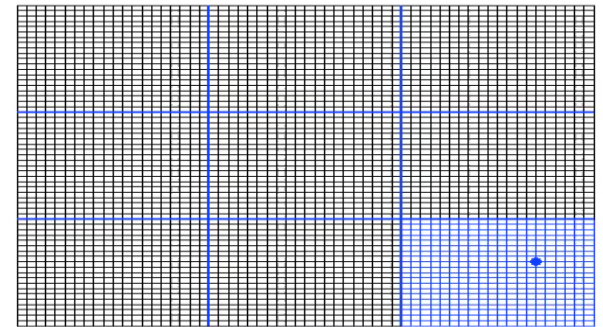
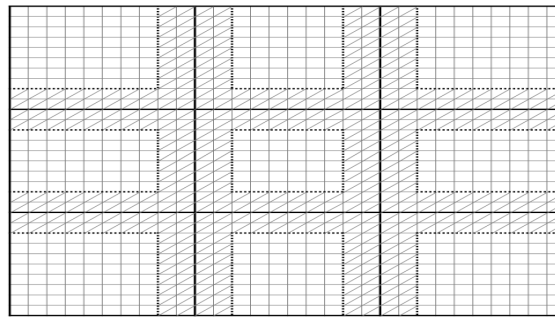
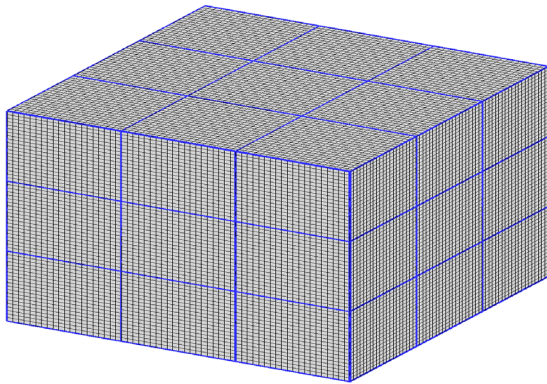
Restricted Kalman filter: select Q with $L = C^* R^{-1}$ and $C_1 = C|_{\Omega_1} = 0$ so that $\Sigma = I$

Constrained Kalman filter: solve Kalman filter over entire spatial domain such that

$$\lambda|_{\Omega_2} = \lambda_2 \quad \text{and} \quad \lambda|_{\Omega_1} = 0$$

OpenMP HT-NODDE-FVM with TVD-RK for Hybrid Estimator Grid Generation and OpenMP

- The computational domain is discretized using $N_{DDX} \times N_{DDY} \times N_{DDZ}$ cuboidal subdomains.
- Each subdomain has the same number of cells with arbitrary uniform $\Delta x, \Delta y, \Delta z$.
- All the loop iterations in each subdomain are handled by a distinct thread with a distinct thread ID number.
- The data of the outermost two layers of cells in each subdomain (the region with solid gray hatched lines) are needed by the adjacent subdomains to calculate the fluxes on the subdomain interfaces when using the TVD scheme and to enforce transmission conditions.
- Since OpenMP is a type of shared-memory parallel implementation, the whole memory can be accessed by each thread, no extra coding is needed to handle the data communication between different threads.



OpenMP HT-NODDE-FVM Method for Hybrid Estimator FVM-TVD Spatial Discretization

- The estimator model equation in each subdomain

$$\begin{aligned}
 - \frac{\partial \hat{C}}{\partial t} &= - \frac{\partial(\hat{C}U)}{\partial X} - \frac{\partial(\hat{C}V)}{\partial Y} - \frac{\partial(\hat{C}W)}{\partial Z} + \frac{\partial}{\partial X} \left(K_{XX} \frac{\partial \hat{C}}{\partial X} \right) + \frac{\partial}{\partial Y} \left(K_{YY} \frac{\partial \hat{C}}{\partial Y} \right) + \frac{\partial}{\partial Z} \left(K_{ZZ} \frac{\partial \hat{C}}{\partial Z} \right) + \\
 - \begin{cases} R(\Theta_s(t), t), & \text{Luenberger Estimator} \\ 0, & \text{Naive Estimator} \end{cases}
 \end{aligned}$$

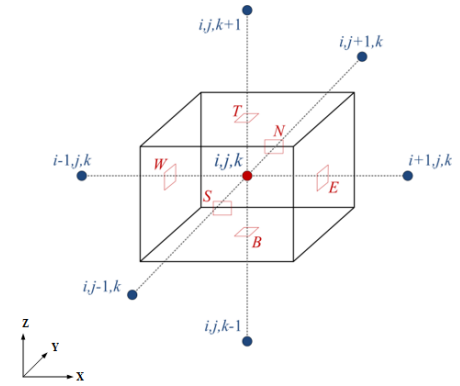
- Integrate over a finite volume $\Omega_{ijk(m)}$ and use Gaussian theorem

$$\begin{aligned}
 - \frac{\partial}{\partial t} \iiint_{\Omega_{ijk(m)}} \hat{C}_{ijk(m)} d\Omega + \oint_{A_{ijk(m)}} \hat{\mathbf{F}}_{ijk(m)} \cdot d\mathbf{A} = \\
 \begin{cases} \iiint_{\Omega_{ijk(m_L)}} R_{ijk(m_L)} d\Omega, & \text{Luenberger Estimator} \\ 0, & \text{Naive Estimator} \end{cases}
 \end{aligned}$$

- The flux vectors are

$$\hat{\mathbf{F}}_{ijk(m)} = \hat{F}_{X,ijk(m)} \mathbf{i} + \hat{F}_{Y,ijk(m)} \mathbf{j} + \hat{F}_{Z,ijk(m)} \mathbf{k} = (\hat{f}_{X,ijk(m)}^A + \hat{f}_{X,ijk(m)}^D) \mathbf{i} + (\hat{f}_{Y,ijk(m)}^A + \hat{f}_{Y,ijk(m)}^D) \mathbf{j} + (\hat{f}_{Z,ijk(m)}^A + \hat{f}_{Z,ijk(m)}^D) \mathbf{k}$$

- Advective components are $\hat{f}_{X,ijk(m)}^A = \hat{C}_{ijk(m)}U$, $\hat{f}_{Y,ijk(m)}^A = \hat{C}_{ijk(m)}V$, $\hat{f}_{Z,ijk(m)}^A = \hat{C}_{ijk(m)}W$
- Diffusive components are $\hat{f}_{X,ijk(m)}^D = -K_{XX} \frac{\partial \hat{C}_{ijk(m)}}{\partial X}$, $\hat{f}_{Y,ijk(m)}^D = -K_{YY} \frac{\partial \hat{C}_{ijk(m)}}{\partial Y}$, $\hat{f}_{Z,ijk(m)}^D = -K_{ZZ} \frac{\partial \hat{C}_{ijk(m)}}{\partial Z}$
- The advective flux \hat{f}_X^A at a face of a finite volume interior to the subdomain or a face of a finite volume at the boundary or between two subdomains is approximated using a TVD scheme of Thuburn (1997) with the limiter function of Kurganov and Tadmor (2020).
- The diffusive flux \hat{f}_X^D at a face of a finite volume interior to the subdomain or a face of a finite volume at the boundary between two subdomains is approximated with central differencing.



OpenMP HT-NODDE-FVM with TVD-RK for Hybrid Estimator

- Enforce flux continuity transmission.
 - $\mathbf{v}_{(p)} \cdot \mathbf{n}_{(p)} \hat{C}_{(p)\Gamma}(\mathbf{r}, t) - K_{(p)} \nabla \hat{C}_{(p)\Gamma}(\mathbf{r}, t) \cdot \mathbf{n}_{(p)} = -(\mathbf{v}_{(q)} \cdot \mathbf{n}_{(q)} \hat{C}_{(q)\Gamma}(\mathbf{r}, t) - K_{(q)} \nabla \hat{C}_{(q)\Gamma}(\mathbf{r}, t) \cdot \mathbf{n}_{(q)}), \forall \mathbf{r} \in \Gamma; \Gamma = \partial\Omega_{(p)} \cap \partial\Omega_{(q)}; p, q = 1, 2, \dots, N_{sd} \text{ and } p \neq q$
- An example, for east cell interface, positive wind speed and diffusion from volume $\Omega_{ijk(p)}$ with $p = m, q = m^E$
 - $\hat{f}_X^A|_{ijk(m)}^E + \hat{f}_X^D|_{ijk(m)}^E = -\hat{f}_X^A|_{i+1,j,k,(m^E)}^W - \hat{f}_X^D|_{i+1,j,k,(m^E)}^W$
- The continuity transmission condition
 - $\hat{C}_{(p)\Gamma}(\mathbf{r}, t) = \hat{C}_{(q)\Gamma}(\mathbf{r}, t), \forall \mathbf{r} \in \Gamma; \Gamma = \partial\Omega_{(p)} \cap \partial\Omega_{(q)}; p, q = 1, 2, \dots, N_{sd} \text{ and } p \neq q$
 - at an interface is also explicitly satisfied since nodal values, if needed, in the FVM are obtained from weighting values of all adjacent cells.
- The integral equation in semi-discrete form
 - $$\frac{\partial \hat{C}_{ijk(m)}}{\partial t} = -\frac{1}{\Omega_{ijk(m)}} \left(\hat{F}_{ijk(m)}^E A_{ijk(m)}^E - \hat{F}_{ijk(m)}^W A_{ijk(m)}^W + \hat{F}_{ijk(m)}^N A_{ijk(m)}^N - \hat{F}_{ijk(m)}^S A_{ijk(m)}^S + \hat{F}_{ijk(m)}^T A_{ijk(m)}^T - \hat{F}_{ijk(m)}^B A_{ijk(m)}^B \right) + \begin{cases} R_{ijk(m_L)}, & \text{Luenberger Estimator} \\ 0, & \text{Naive Estimator} \end{cases}$$
- The output injection term in discretized form,
 - $$R_{ijk(m_L)} = \begin{cases} 0, & \Theta_s(t) \notin \Omega_{ijk(m_L)} \\ \Omega_{ijk(m_L)} \Lambda [\mathcal{C}(\Theta_s(t), t) - \hat{\mathcal{C}}(\Theta_s(t), t)], & \Theta_s(t) \in \Omega_{ijk(m_L)} \end{cases}$$

OpenMP HT-NODDE-FVM Method for Hybrid Estimator

- Mapping of the entire set of $N_{X(m)} \times N_{Y(m)} \times N_{Z(m)}$ finite volumes in each subdomain $i, j, k \in \Omega_{(m)}$ to a 1D vector
 - $np_{(m)} = np_{(m)}(i, j, k) = i + (j - 1)N_{X(m)} + (k - 1)N_{X(m)}N_{Y(m)}$
 - $i = 1, 2, \dots, N_{X(m)} ; j = 1, 2, \dots, N_{Y(m)} ; k = 1, 2, \dots, N_{Z(m)}$
- The semi-discrete equations for the estimator model can be written as
 - $$\frac{\partial \hat{c}_{np(m)}}{\partial t} = - \frac{1}{\Omega_{np(m)}} \left(\hat{F}_{np(m)}^E A_{np(m)}^E - \hat{F}_{np(m)}^W A_{np(m)}^W + \hat{F}_{np(m)}^N A_{np(m)}^N - \hat{F}_{np(m)}^S A_{np(m)}^S + \hat{F}_{np(m)}^T A_{np(m)}^T - \hat{F}_{np(m)}^B A_{np(m)}^B \right) + \begin{cases} R_{np(m_L)}, & \text{Luenberger estimator} \\ 0, & \text{Naive estimator} \end{cases}$$
- We set
 - $\hat{\mathbf{x}}_{(m)}(t) = \hat{\mathbf{C}}_{(m)}(X, Y, Z, t)$ for the estimated concentration state
 - $\mathbf{P}_{RHS(m)}$ for the right-hand side of the semi-discrete equations
- Integrated with respect to time using a 4th order Runge-Kutta method
 - $\hat{\mathbf{x}}_{(m)}^{(k)} = \hat{\mathbf{x}}_{(m)}^{(l)} + \alpha_k \Delta t \mathbf{P}_{RHS(m)}^{(k-1)} ; k = 1, \dots, 4; \alpha_1 = 0; \alpha_2 = \frac{1}{2}; \alpha_3 = \frac{1}{2}; \alpha_4 = 1$
 - $\hat{\mathbf{x}}_{(m)}^{(l+1)} = \hat{\mathbf{x}}_{(m)}^{(l)} + \Delta t \sum_{k=1}^4 \beta_k \mathbf{P}_{RHS(m)}^{(k)} ; \beta_1 = 1/6; \beta_2 = 2/6; \beta_3 = 2/6; \beta_4 = 1/6$
- The time step Δt must satisfy the stability conditions obtained by Von Neumann stability analysis
 - $$\Delta t \leq \Delta t_{max} = \min(\Delta t_1, \Delta t_2, \Delta t_3) = \min \left\{ \frac{1}{\frac{U}{\Delta X} + \frac{V}{\Delta Y} + \frac{W}{\Delta Z} + 2 \left[\frac{K_{XX}}{(\Delta X)^2} + \frac{K_{YY}}{(\Delta Y)^2} + \frac{K_{ZZ}}{(\Delta Z)^2} \right]}, \frac{1}{2 \left[\frac{K_{XX}}{(\Delta X)^2} + \frac{K_{YY}}{(\Delta Y)^2} + \frac{K_{ZZ}}{(\Delta Z)^2} \right]}, \frac{2 \left[\frac{K_{XX}}{(\Delta X)^2} + \frac{K_{YY}}{(\Delta Y)^2} + \frac{K_{ZZ}}{(\Delta Z)^2} \right]}{\left(\frac{U}{\Delta X} \right)^2 + \left(\frac{V}{\Delta Y} \right)^2 + \left(\frac{W}{\Delta Z} \right)^2} \right\}$$
- The UAV trajectory equation provides $\mathbf{r}_s(t) = [X_s^d \ Y_s^d \ Z_s^d]^T$ by integration using the fourth order Runge-Kutta method,
 - $\mathbf{r}_s^{(k+1)} = \mathbf{r}_s^{(l+1)} + \alpha_k \Delta t \mathbf{N}_{RHS}^{(k-1)} ; k = 1, \dots, 4;$
 - $\mathbf{r}_s^{(l+1)} = \mathbf{r}_s^{(l)} + \Delta t \sum_{k=1}^4 \beta_k \mathbf{N}_{RHS(m)}^{(k)}$

Simulation Parameters

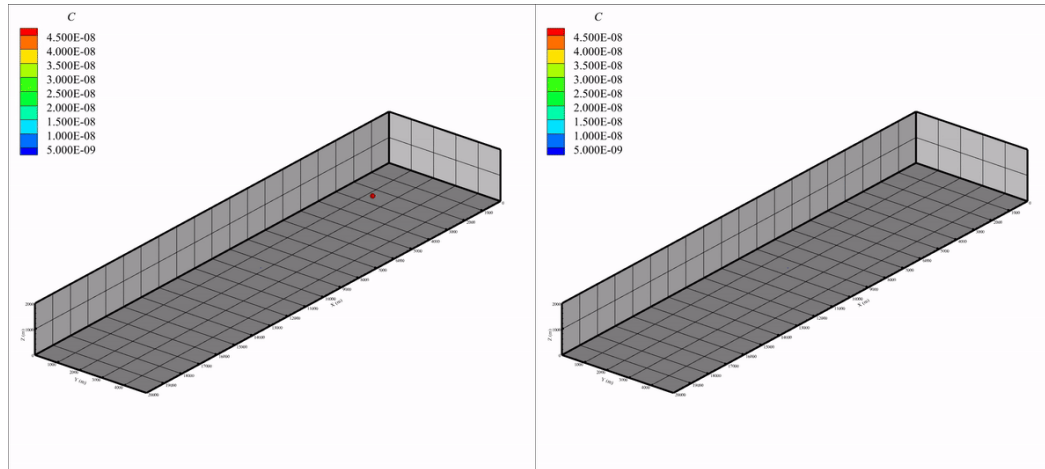
- SAV with 6DOF Dynamics (Aerosonde)

Computational Conditions			Intruder and SAV Parameters			Aerosonde Dynamical Parameters		
Domain length (km)	L_X, L_Y, L_Z	20, 5, 2	Initial intruder location (km)	X_c, Y_c, Z_c	4.1, 2.5, 1 or 0.1, 2.5, 1	Mass (kg)	M	13.5
Number of cells	N_X, N_Y, N_Z	300, 75, 30	Intruder velocity (m/s)	v_X, v_Y, v_Z	50, 0, 0	Wing planform area (m ²)	S	0.55
Number of subdomains	$N_{DDX}, N_{DDY}, N_{DDZ}$	3, 3, 3	Initial SAV location (km)	X_{s0}, Y_{s0}, Z_{s0}	10.4 or 12, 2.5, 0.55	Wingspan (m)	b	2.9
X-direction wind speed (m/s)	$U = \begin{cases} U_r \left(\frac{Z}{Z_r}\right)^p, & Z < Z_r \\ U_r, & Z \geq Z_r \end{cases}$	$Z_r = 500$ m $U_r = 10 - 20$ m/s or $U_r = 5 - 10$ m/s $p = 0.4$	Helix center and radius of SAV patrolling (km)	$X_{scc}, Y_{scc}, Z_{scc}, r$	8 or 9.6, 2.5, 0.55, 2.4	Ground speed range (m/s)	V_g	27 – 33
Eddy diffusivities (m ² /s)	K_{XX}, K_{YY}, K_{ZZ}	100, 100, 40	Helix pitch (rad)	$pitch$	120π	Maximum thrust (N)	T_{max}	50
Sensor threshold (kg/m ³)	C_{min}	1×10^{-9}	SAV patrolling linear velocity (m/s)	v_s	30	Maximum bank angle (rad)	ϕ_{max}	0.5
Estimation gain	Λ	5×10^{-6}	Guidance gains	k_X, k_Y, k_Z	21, 21, 4	Lift coefficient in stall condition	C_{Lmax}	1.5
Time steps (s)	$\Delta t, \Delta t_s$	1, 1				Oswald efficiency factor	e	0.9
						Parasitic drag coefficient	C_{Dp}	0.0437
						Initial patrolling flight path angle and course angle (rad)	γ_0, χ_0	0.0524, 1.5708
						Maximum bank rate (rad/s)	$\dot{\phi}_{max}$	0.4
							b_{V_g}	1
							b_Y	0.5
							b_X	0.5
							b_ϕ	1

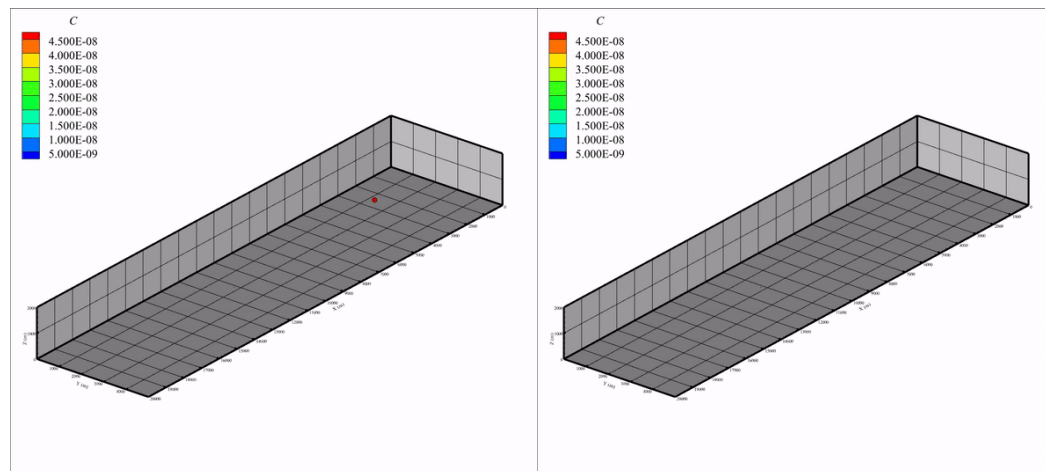
Simulation Results

SAV with 6DOF Dynamics (Aerosonde)

Wind speed = 10 m/s



Wind speed = 20 m/s



Experimental verification

Equipment supported by AFOSR DURIP FA9550-22-1-0138 ``*Testbed for Multimodal Sensor Configuration, Real-Time Estimation, and Optimal Control in Autonomous Systems*''

- Use projectors (Optoma ZH450ST) to project various spatial fields onto floor
- Colors represent different field intensities
- Use mobile robot (Turtlebot 4 lite robot) with color cameras (Logitech C920x webcam) to read field intensity
- Use Raspberry pi 4 model B, for image processing
- Base station is a Dell Latitude 7330 Rugged Extreme Laptop

Experimental verification

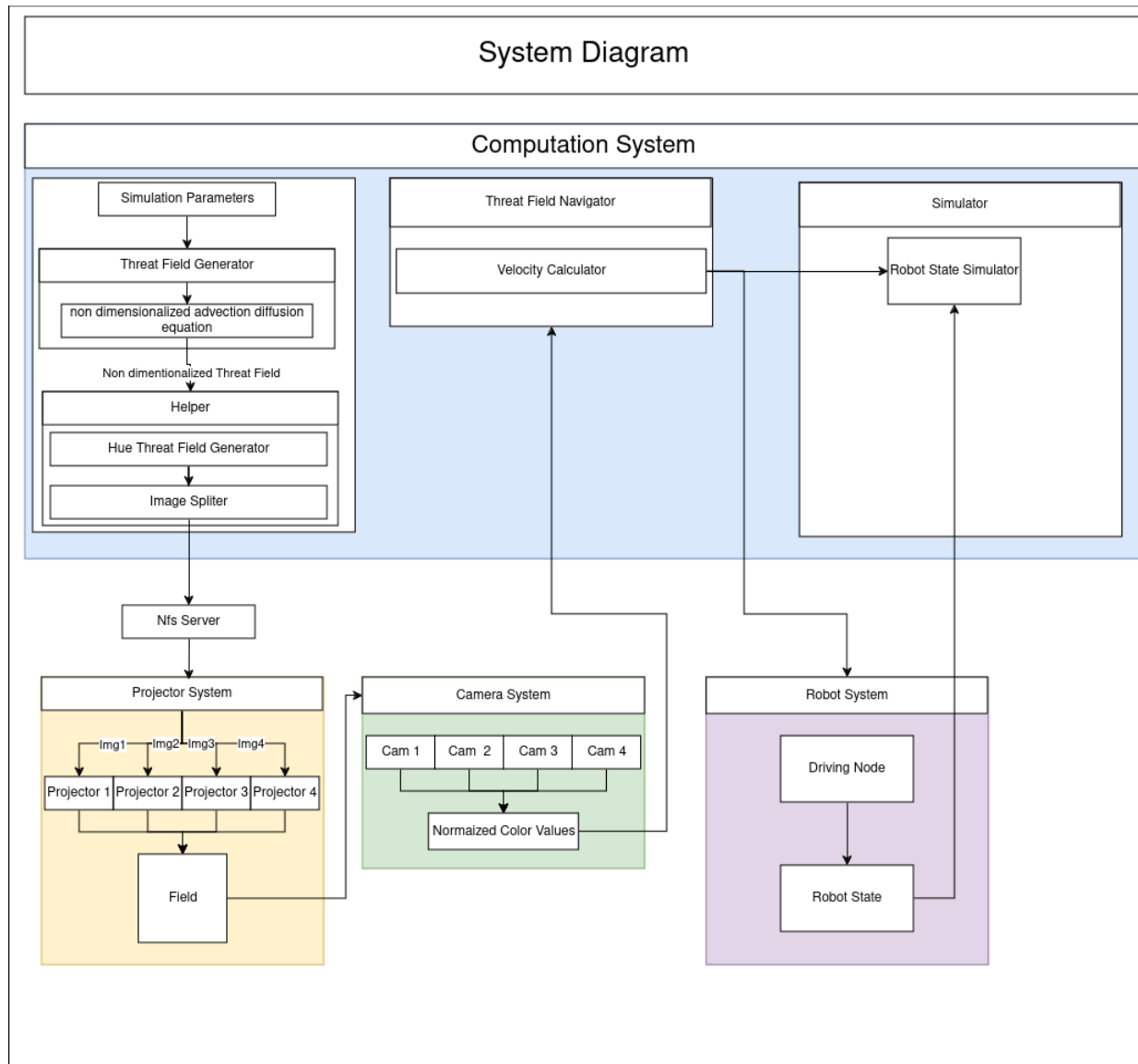
Robot Communication

- robot communicates with the base station by broadcasting the current robot state in a ROS topic and receiving the target velocity and orientation from the base station. These ROS topics work by establishing a topic connection across a network using an HTTP based protocol. Specifically, it uses the protocol TCPROS which is a persistent, stateful TCP/IP socket connection. For our system this is done through communicating with our local Wi-Fi created by a NETGEAR router, where all the devices for the experiment are connected.

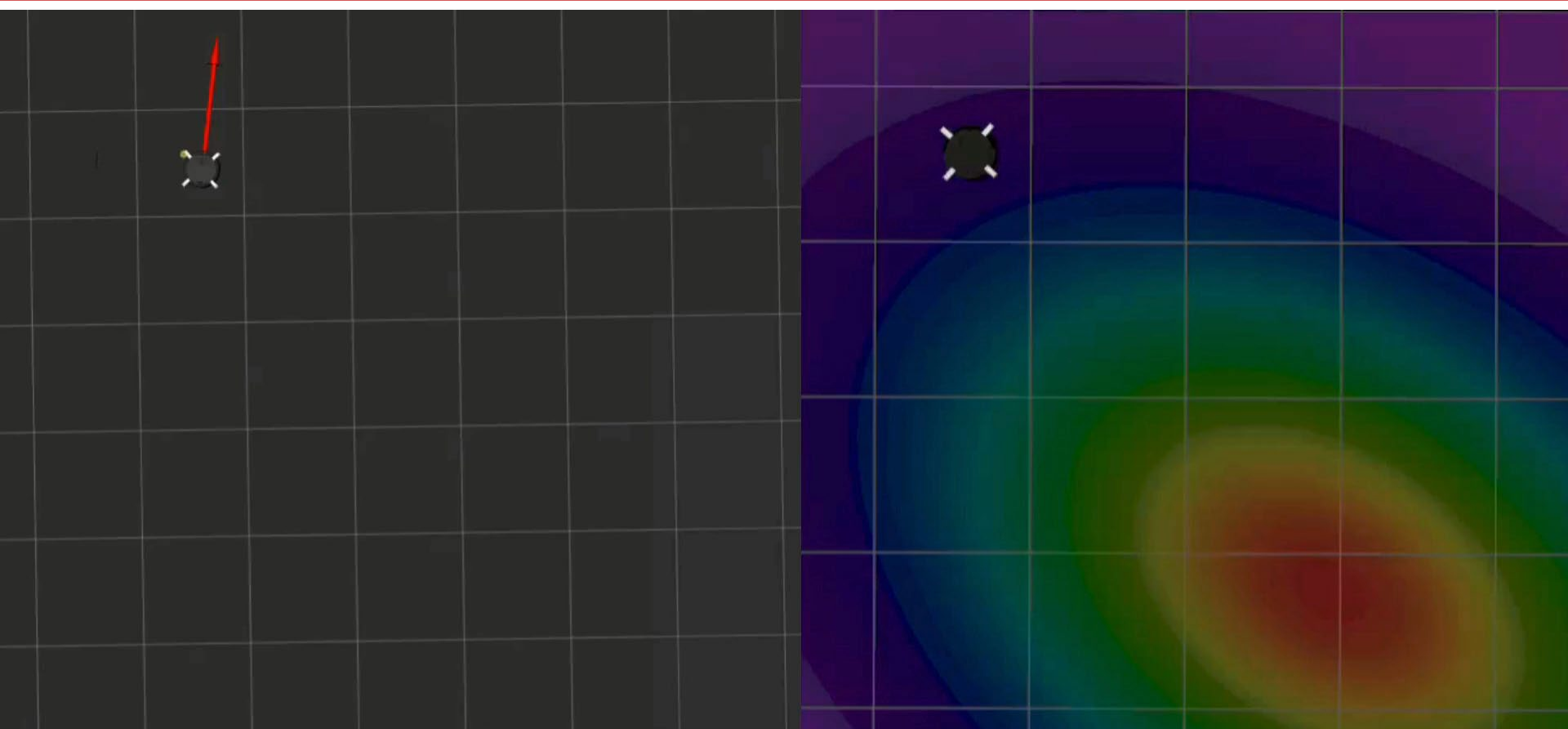
Robot Operating System is ROS2.

First stage: use gradient-based navigation to find maximum of spatiotemporally varying field

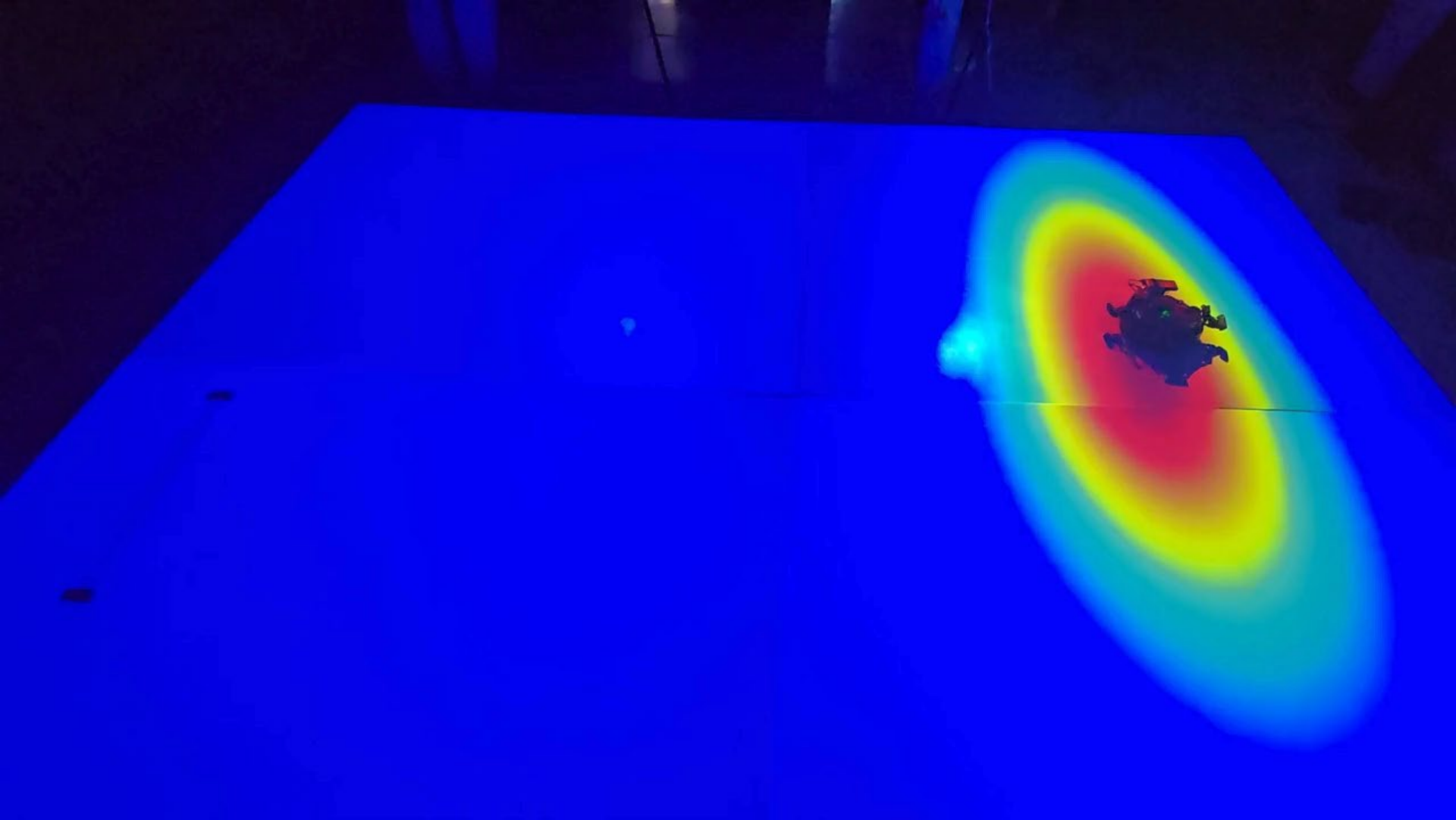
Experimental verification



Experimental verification: ROS output to real work space



Experimental verification



Timeline

tasks	Yr 1	Yr 2	Yr 3
T1: Kalman Filter over inner domain with pointwise measurement	ECC 25, CDC 25, Journal 1(th.) Journal 2 (impl.)		
T2: Kalman Filter over inner domain using other sensor distributions		ACC 26 CDC 26 Journal 3 (th) Journal 4 (impl.)	
T3: Hybrid DD using sensor network	SciTech 25		CDC 27 SciTech 28 Journal 5 (th.) Journal 6 (impl.)

References

1. “Hybrid domain decomposition filters for parabolic spatially distributed processes”, Michael A. Demetriou and Weiwei Hu, ACC2019.
2. “Domain Decomposition for a Heterogeneous State Estimation of a Plume Field with a Moving Sensor in 2D”, Xin Tian, Michael A. Demetriou, and Nikolaos Gatsonis, 2020 AIAA SciTech.
3. “Hybrid domain decomposition filters for advection-diffusion PDEs with mobile sensors”, Weiwei Hu, M. A. Demetriou, Xin Tian, N. A. Gatsonis, *Automatica*, 2022.
4. “A Heterogeneous Non-Overlapping Domain Decomposition Explicit Finite Volume Method for a Real-Time Hybrid Process-State Estimator of 3D Unsteady Advection-Diffusion Fields”, Nikolaos A. Gatsonis, Xin Tian, Michael A. Demetriou, John Burns *Journal of Computational Physics*, 2022.
5. “Gradient Evaluation in Spatiotemporal Plumes Using a Sensing Autonomous Vehicle”, Nicholas B. Orlovsky, Tyler Lizotte, Michael A. Demetriou and Nikolaos A. Gatsonis, accepted to 2025 AIAA SciTech.
6. “Simplified Kalman filter design for a Heterogeneous State Estimation of a Plume Field with a Moving Sensor”, Tyler Lizotte, Michael A. Demetriou, Nikolaos A. Gatsonis, to be submitted to ECC25.
7. “Hybrid domain decomposition restricted Kalman filters for PDEs”, under preparation (Journal 1, theoretical).

Epilogue

Questions