

Using mobile sensor density to approximate state feedback controllers for a class of PDEs[★]

Michael A. Demetriou^{*} Weiwei Hu^{**}

^{*} Aerospace Engineering Department, WPI, Worcester, MA 01609, USA
(e-mail: mdemetri@wpi.edu).

^{**} Department of Mathematics, University of Georgia, Athens, GA 30602
USA (e-mail: Weiwei.Hu@uga.edu)

Abstract: This work is concerned with the use of mobile sensors to approximate and replace the full state feedback controller by static output feedback controllers for a class of PDEs. Assuming the feedback operator associated with the full-state feedback controller admits a kernel representation, the proposed optimization aims to approximate the inner product of the kernel and the full state by a finite sum of weighted scalar outputs provided by the mobile sensors. When the full state feedback operator is time-dependent thus rendering its associated kernel time-varying, the approximation results in moving sensors with time-varying static gains. To calculate the velocity of the mobile sensors within the spatial domain, the time-varying kernel is set equal to the sensor density and thus the solution to an associated advection PDE reveals the velocity field of the sensor network. To obtain the speed of the finite number of sensors, a domain decomposition based on a modification of the Centroidal Voronoi Tessellations (μ -CVT) is used to decompose the kernel into a finite number of cells, each of which contains a single sensor. A subsequent application of the μ -CVT on the velocity field provides the individual sensor speeds. The nature of this μ -CVT ensures collision avoidance by the very structure of the kernel decomposition into non-intersecting cells. Numerical simulations are provided to highlight the proposed sensor guidance.

Copyright © 2022 The Authors. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Keywords: Distributed parameter systems; sensor selection; sensor guidance; sensor density; static output feedback; control approximation; kernel approximation; computational geometry.

1. INTRODUCTION

Facing the conflict between (a) the superb performance afforded by full state feedback and (b) the reduced computational load combined with simplified controller architecture provided by static output feedback (proportional controllers), this paper attempts to combine the positive attributes of these two distinct controller architectures. The premise is that the state operator associated with full state feedback for a class of PDEs, admits a kernel representation, King (1995, 2000). Using the controller expression that is computed as the inner product of the kernel and the unavailable process state, a geometric control method is used to approximate the full state feedback controller by a finite sum of weighted output measurements. All that remains in this full-state feedback approximation is to find (i) the sensor locations and (ii) the associated static gains (proportional controllers). The first part is attained by approximating the feedback kernel by a finite sum of output sensor distributions. The enabling scheme is based on a modification of the Centroidal Voronoi Tessellations (μ -CVT) that decomposes the feedback kernel into a finite number of disjoint cells (domain decomposition) with the property that each cell has the same area under the feedback kernel. Decomposing the kernel into a number of equiareal cells enables one to place a sensor in each cell with the property that each sensor has the same level of “spatial authority” with respect to the kernel. Once the sensor positions within the spatial domain are identified, their associated gains

are easily computed via a least-squares approximation. This approach bears similarities with the sampled-in-space actuation and sensing presented in Pisano and Orlov (2017), since the time-varying feedback kernel is approximated by sampled-in-space sensor information for each time.

When the state feedback is time-varying, as for example when one uses a finite-horizon LQR cost for controller design, or when the process operators are time-varying, then the associated kernel is rendered time-varying. The approximation of the full state feedback control law must be implemented continuously since both the sensor locations and their associated static gains are time-varying. One approach is to use the above μ -CVT method to decompose the spatial domain at each time and compute the time-dependent gains at each time. Such a computational feat can easily be achieved via the use of a fast processor and a high-bandwidth data-acquisition-and-control hardware. This approach is particular to the number of sensors used. To generalize this for any number n_s of sensors, a modification is considered here. Instead of declaring the location of the mobile sensors at each time via the approximation of the time-varying kernel, we propose to operate on the sensor density to extract the velocity field of the sensor network. Thus, instead of commanding the position of each sensor at each time, the proposed work provides the speed of each sensor.

The sensor speed is obtained by substituting the sensor density in the advection PDE representing the evolution of the sensor network density, by the time-varying kernel. The solution of this PDE provides the velocity of the sensor network. All is left at this stage, is to identify the speed of each sensor in the

[★] The first author acknowledges financial support from NSF-CMMI grant # 1825546. The second author acknowledges financial support from NSF-DMS grants # 2111486 and # 1813570.

network. This is achieved by applying the decomposition used for the feedback kernel, to the velocity field. By its nature, this μ -CVT based decomposition ensures collision avoidance, thus simplifying the mobile sensor guidance.

Operating on the sensor density, a similar approach considers the advection PDE representing the evolution of the sensor density, but formulates the network velocity as a bilinear control tracking problem whereby one selects the network velocity as the control signal in order to ensure that the sensor density tracks the time-varying feedback kernel. The highlights of the three kernel approximation schemes are given below.

- (1) Use a computational geometry scheme to decompose the feedback kernel into n_s cells and place a sensor in each cell. The position of each sensor and the static gain is provided at each time t .
- (2) Set the kernel equal to the solution of a density advection PDE and solve for the velocity of the associated sensor network. The speed of each sensor and the static gain is provided at each time t .
- (3) Solve the bilinear tracking control problem in the density advection PDE. The control signal is the velocity function and the sensor density is forced to track the feedback kernel. The speed of each sensor and the static gain is provided at each time t .

This paper presents the three approximation schemes that allow one to replace a full-state feedback controller by the weighted sum of measurements provided by mobile sensor network. Section 2 formulates the problem for 1D PDEs. Section 3 presents the three schemes to allow one to replace a full-state feedback controller by a static feedback where both the gains and the sensor locations are time-varying. Numerical results showcasing one approach are presented in Section 4 and conclusions with future extensions follow in Section 5.

2. PROBLEM FORMULATION

The class of PDEs represented by this framework encompasses advection-diffusion equations primarily in 1D spatial domains with the 2D spatial domain case being easily extendable. Such PDEs are abstractly written as evolution equations

$$\dot{x}(t) = \mathcal{A}x(t) + \mathcal{B}u(t), \quad x(0) = x_0 \in \text{dom}(\mathcal{A}), \quad (1)$$

where the state and input operators are $\mathcal{A} \in L(\mathcal{V}, \mathcal{V}^*)$ and $\mathcal{B} \in L(U, \mathcal{V}^*)$, respectively. To account for unbounded input and state operators, the Gelfand triple with $\mathcal{V} \hookrightarrow \mathcal{H} \hookrightarrow \mathcal{V}^*$ is considered with the embeddings dense and continuous, Showalter (1977). The input space $\mathcal{U}$ is considered to be a finite dimensional Euclidean space, Curtain and Zwart (1995). Since a single actuator is considered here, then $\mathcal{U} = \mathbb{R}^1$.

For the above system, it is assumed that a suitable controller can be designed to satisfy certain performance objectives. Such a controller, assumed to have a time-varying operator gain, is

$$u(t) = -\mathcal{K}(t)x(t). \quad (2)$$

The time variation for such a feedback operator gain may come from either a time varying input operator or the one associated with a finite horizon linear quadratic cost functional

$$J = \int_0^T \langle x(\tau), Qx(\tau) \rangle_X + u^T(\tau) \mathcal{R}u(\tau) d\tau. \quad (3)$$

As summarized in Demetriou (2021), such a time-varying feedback operator is obtained from the solution to an associated Operator Differential Riccati Equation

$$-\dot{\mathcal{P}} = \mathcal{A}^* \mathcal{P} + \mathcal{P} \mathcal{A} - \mathcal{P} \mathcal{B}(t) R^{-1} \mathcal{B}^*(t) \mathcal{P} + Q, \quad (4)$$

where for simplicity, a zero terminal condition is assumed with $\mathcal{P}(T) = 0$. In this case, one can express the feedback operator in terms of the above Riccati equation $\mathcal{K}(t) = R^{-1} \mathcal{B}^*(t) \mathcal{P}(t)$.

In the absence of full state information, the controller (2) cannot be implemented. The alternatives are to *either* produce a state estimate $\hat{x}(t)$ by employing a state estimator for the process (1) supplemented with sensor measurements

$$y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_{n_s}(t) \end{bmatrix} = \begin{bmatrix} C_1 x(t) \\ \vdots \\ C_{n_s} x(t) \end{bmatrix} = Cx(t), \quad (5)$$

as provided by n_s sensors, *or* implement an appropriately designed reduced order controller that uses the measurements (5).

The latter is considered here, as a means to minimize the computational costs associated with the implementation of a state estimator and additionally as a means to reduce the complexity of the controller architecture. The basic elements of such an alternate controller for a system (1) employing a controller (2) with a time varying feedback gain are summarized below

- (i) Assume that the time-varying feedback operator admits a time-varying feedback kernel representation.
- (ii) Replace the controller (2) by its static output feedback controller approximation

$$u_s(t) = - \sum_{i=1}^{n_s} \gamma_i(t) y_i(t) \quad (6)$$

where $y_i(t)$, $i = 1, \dots, n_s$ denote the sensor measurements in (5) and $\gamma_i(t)$ are the corresponding time-varying gains.

- (iii) Given that the feedback operator is available, provide a solution to the gain approximation

$$(\mathcal{K}(t) - \Gamma(t)C)x(t) \approx 0, \quad (7)$$

in order to obtain at each time t , the time varying static gains $\gamma_i(t)$, $i = 1, \dots, n_s$.

To improve and to perhaps ensure that the approximation (6) is made feasible, an added level of optimization is incorporated in element (ii) above to allow the sensor measurements be delivered by mobile sensors and thus (7) is replaced by

- (ii)' Given that the feedback operator is available, provide a solution to the gain approximation

$$(\mathcal{K}(t) - \Gamma(t)C(t))x(t) \approx 0, \quad (7')$$

in order to obtain at each time t , the time varying static gains $\gamma_i(t)$, $i = 1, \dots, n_s$ and the sensor locations associated with the time-varying output operator $C(t)$ given in (5).

To address the above, we consider an advection PDE in 1D and formulate the optimization of both the static gains and the time-varying sensor locations in weak form. The advection PDE is

$$\begin{aligned} x_t(t, \xi) &= (a(\xi)x_t(t, \xi))_\xi + a_2 x_\xi(t, \xi) \\ &\quad + a_3 x(t, \xi) + b(t, \xi)u(t) \\ y(t) &= \begin{bmatrix} y_1(t; \xi_1(t)) \\ \vdots \\ y_{n_s}(t; \xi_{n_s}(t)) \end{bmatrix} = \begin{bmatrix} \int_0^\ell c_1(\xi, \xi_1(t))x(t, \xi) d\xi \\ \vdots \\ \int_0^\ell c_{n_s}(\xi, \xi_{n_s}(t))x(t, \xi) d\xi \end{bmatrix}, \end{aligned} \quad (8)$$

where $c_i(\xi; \xi_i(t))$ denotes the i^{th} sensor spatial distribution at the location $\xi_i(t) \in [0, \ell]$. The explicit dependence of the sensor

distributions on time, via the time-varying locations or centroids $\xi_i(t)$, is made to account for the sensor motion within the domain $\Omega = (0, \ell)$. Thus the sensor motion is fully encompassed in the time variation of the centroids $\xi_i(t)$. Dirichlet boundary conditions $x(t, 0) = x(t, \ell) = 0$ are assumed in (8). The function $b(t, \xi)$ is the distribution of the actuating device and accounts for a possibly moving actuator. The coefficients $a(\xi)$, a_2 , a_3 are associated with the elliptic operator.

To simplify the analysis, a rather straightforward assumption is made on the sensing devices and considers a network of sensing devices that are identical and differing only on the position of their centroid within the spatial domain Ω .

Assumption 1. (uniform sensor network). The sensor network comprised of the n_s sensing devices contains identical sensors

$$c_i(\xi, \xi_i(t)) = c(\xi, \xi_i(t)), \quad i = 1, \dots, n_s. \quad (9)$$

The other assumption ensures that the proposed computational geometry approach for the approximation (7') is feasible.

Assumption 2. (kernel representation). The time varying feedback gain operator is assumed to admit a kernel representation. For the PDE (8), this representation becomes

$$\mathcal{K}(t)\phi = \int_0^\ell k(t, \xi)\phi(\xi) d\xi, \quad \forall \phi \in H_0^1(0, \ell). \quad (10)$$

This is valid for parabolic systems with any bounded control for 1-3D and for Neumann B.C.'s for 1-2D, King (1995, 2000).

Problem Statement: Using Assumptions 1,2, the approximation (7') for the scalar control case, viewed in weak form, becomes

$$\int_0^\ell \left(k(t, \xi) - \sum_{i=1}^{n_s} \gamma_i(t) c(\xi, \xi_i(t)) \right) \phi(\xi) d\xi \approx 0, \quad (11)$$

for all $\phi \in H_0^1(0, \ell)$. The optimization problem seeks to produce the static gains $\gamma_i(t)$ and the associated sensor locations $\xi_i(t)$ that minimize the difference in the above integrand in an appropriate sense. Once the optimal gains $\gamma_i^{opt}(t)$ and optimal locations $\xi_i^{opt}(t)$ for each time t are produced, then the control

$$u(t) = - \int_0^\ell k(t, \xi) x(t, \xi) d\xi \quad (12)$$

which corresponds to (2) is replaced by the approximate static output feedback controller

$$\begin{aligned} u_s(t) &= - \sum_{i=1}^{n_s} \gamma_i^{opt}(t) \int_0^\ell c(\xi, \xi_i^{opt}(t)) x(t, \xi) d\xi \\ &= - \sum_{i=1}^{n_s} \gamma_i^{opt}(t) y_i(t; \xi_i^{opt}(t)) \end{aligned} \quad (13)$$

Joint gain and sensor location optimization: The optimization described by (11) is formally expressed as

$$\min_{\phi \in H_0^1(0, \ell)} \left| \int_0^\ell \left(k(t, \xi) - \sum_{i=1}^{n_s} \gamma_i(t) c(\xi, \xi_i(t)) \right) \phi(\xi) d\xi \right|^2. \quad (14)$$

Equation (14) seeks ways to approximate the feedback kernel for each time $t \in [0, T]$ so that the difference is minimized. Expanding the approach presented in Demetriou (2021), we present three different methods to solve the optimization (14):

- (1) (*Kernel decomposition*) using computational geometry approaches: In this case, a modification of the Centroidal Voronoi Tessellations (μ -CVT) is used to partition the kernel into n_s cells $I_i(t)$, each containing a sensor at location

$\xi_i(t) \in I_i(t)$ and use a least-squares approach to solve for the associated feedback gains $\gamma_i(t)$. The outputs here are the locations $\xi_i(t)$ and the gains $\gamma_i(t)$ for each $t \in [0, T]$.

- (2) (*Sensor network density propagation:*) In this case, the kernel is viewed as equivalent, up to a constant, to the sensor network density. The solution to the associated advection PDE that propagates the sensor density forward in time, yields the network velocity. Once the sensor network velocity is obtained, any decomposition (partitioning) of the velocity will yield the speed $\dot{\xi}_i(t)$ and direction of each sensor. Subsequently, the same decomposition provides the associated feedback gains. The outputs here are the speeds $\dot{\xi}_i(t)$ and the gains $\gamma_i(t)$ for each $t \in [0, T]$.
- (3) (*Bilinear tracking control:*) In this case, the advection PDE describing the sensor density is considered and the goal is to design a velocity controller for the bilinear control problem to ensure that the state of the density PDE tracks the feedback kernel. The outputs here are the speeds $\dot{\xi}_i(t)$ and the gains $\gamma_i(t)$ for each $t \in [0, T]$.

3. SOLUTION TO OPTIMIZATION (14)

The approaches above are detailed here. The first two are taken from Demetriou (2021) and are included here for completeness, whereas the third one represents new work by the authors.

3.1 Kernel decomposition

The approach here follows directly from Demetriou (2021). The idea is to decompose the feedback kernel $k(t, \xi)$ for each time t into n_s equi-areal cells. Each cell will have an area equal to $1/n_s$ of the total area of the kernel $k(t, \xi)$. Then in each cell, place a sensor $\xi_i(t)$ so that in turn decomposes the cell into two equi-areal subcells. The algorithm presented in Demetriou (2021) and adapted for time-variation of the kernel is summarized. A further simplification assumes that the sensor distributions in (9) are described by the Dirac delta functions

$$c(\xi, \xi_i(t)) = \delta(\xi - \xi_i(t)), \quad i = 1, \dots, n_s.$$

Sensor position $\xi_i(t)$ in each time

- (a) *Time varying domain decomposition:* At each time $t \in [0, T]$, partition the domain Ω in n_s cells $I_i(t) \in \bar{\Omega}$ with the property $I_i(t) \cap I_j(t) = \emptyset$ and $\bigcup_{i=1}^{n_s} I_i(t) = \bar{\Omega}$ so that at each cell $I_i(t)$, the area of the kernel $k(t, \xi)$ satisfies

$$\int_{I_i(t)} k(t, \xi) d\xi = \frac{\int_{\Omega} k(t, \xi) d\xi}{n_s} = \left(\frac{A(t)}{n_s} \right), \quad i = 1, \dots, n_s. \quad (15)$$

- (b) *Sensor placement:* Place a sensor $\xi_i(t)$ in each cell $I_i(t)$, using the method in step (a) which ensures that the location $\xi_i(t)$ in cell $I_i(t)$ is such that it subsequently divides the cell into two subcells $I_i^a(t) = I_i^a(t) \cup I_i^b(t)$ of equal areas of the kernel, with each being equal to $0.5A(t)/n_s$.

As presented in Demetriou (2021), there is a simple graphical approach for the 1D case. For each time $t \in [0, T]$, one defines the normalized kernel $\bar{k}(t, \xi)$ and the associated normalized area function under the kernel via

$$\bar{k}(t, \xi) = \frac{k(t, \xi)}{\max_{\xi \in \Omega} k(t, \xi)}, \quad \bar{A}(t, \xi) = \frac{A(t, \xi)}{A(t, \ell)}, \quad A(t, \xi) = \int_0^\xi k(t, \xi) d\xi.$$

For each time $t \in [0, T]$, the area function at $\xi = \ell$, $A(t, \ell)$ is the total area under the kernel at that time and represents the

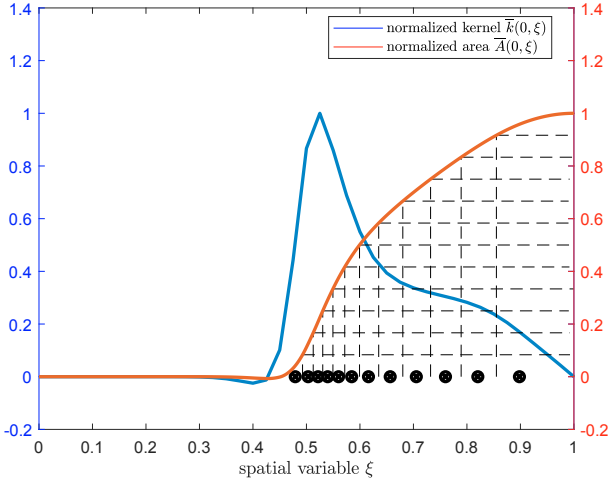


Fig. 1. Normalized feedback kernel $\bar{k}(t, \xi)$ and normalized areal function $\bar{A}(t, \xi)$ for a given time $t \in [0, T]$. The spatial domain Ω is partitioned into $n_s = 12$ equiareal cells I_i with $\int_{I_i} k(t, \xi) d\xi = \frac{1}{n_s} \int_0^\ell k(t, \xi) d\xi$. The cell coordinates (grid points) are found as the abscissae points on the normalized areal function $\bar{A}(t, \xi)$ having ordinates $i/12$.

fraction of the area under the kernel up to the spatial point $\xi \in \Omega$. The identity $\bar{A}(t, \ell) = 1$ trivially follows. When the normalized area $\bar{A}(t, \xi)$ is plotted, then the cells $I_i(t)$ are easily identified by first identifying the ordinate points $1/n_s$ of $\bar{A}(t, \xi)$. The corresponding abscissae points produce the boundaries of the cells $I_i(t)$. They partition Ω into n_s equiareal cells under the kernel. In fact, one does not need to use the graph, but instead obtain the points that partition the spatial domain via

$$\xi_i(t) : \int_0^{\xi_i(t)} k(t, \xi) d\xi = \frac{(2i-1)}{2} \frac{A(t, \ell)}{n_s}, \quad i = 1, \dots, n_s.$$

Figure 1 showcases the graphical method for $n_s = 12$. The normalized areal function $\bar{A}(t, \xi)$ is decomposed into $n_s = 12$ cells with uniform grid points $i/n_s, i = 1, \dots, n_s$. The abscissae points provide the grid points on the ξ -axis that decompose the normalized kernel $\bar{k}(t, \xi)$ into n_s cells that have equal areas.

Feedback gains $\gamma_i(t)$ in each time

The computation of the gains $\gamma_i(t)$ is based on the finite dimensional representation of (8). Their calculation uses the fact that the time-varying kernel has the representation

$$k(t, \xi) = \sum_{j=1}^N k_j(t) \phi_j(\xi), \quad \forall \phi_j \in H_0^1(0, \ell). \quad (16)$$

This approximation is replaced in the optimization (14), viewed in weak form for the test functions $\phi \in H_0^1(0, \ell)$, and which yields the resulting finite dimensional representation

$$\int_0^\ell \sum_{j=1}^N k_j(t) \phi_j(\xi) \phi(\xi) d\xi \approx \sum_{i=1}^{n_s} \gamma_i(t) \int_{I_i(t)} \delta(\xi - \xi_i(t)) \phi(\xi) d\xi.$$

Replacing the test function ϕ by the trial functions $\phi_m(\xi)$, we arrive at the finite dimensional system

$$\int_0^\ell \sum_{j=1}^N k_j(t) \phi_j(\xi) \phi_m(\xi) d\xi \approx \sum_{i=1}^{n_s} \gamma_i(t) \int_{I_i(t)} \delta(\xi - \xi_i(t)) \phi_m(\xi) d\xi$$

for $\phi_m \in H_0^1(0, \ell)$, $m = 1, \dots, N$. In matrix form it becomes

$$M\kappa(t) = \Phi(\xi_s(t))\Gamma(t) \quad (17)$$

where the N -dimensional mass matrix M is

$$[M]_{jm} = \int_0^\ell \phi_j(\xi) \phi_m(\xi) d\xi, \quad j, m = 1, \dots, N.$$

The time-varying kernel coefficients vector $\kappa(t)$ and the corresponding time-varying static gains vector are given by

$$\Gamma(t) = [\gamma_1(t) \dots \gamma_{n_s}(t)]^T, \quad \kappa(t) = [k_1(t) \dots k_N(t)]^T.$$

The matrix $\Phi(\xi_s(t))$ denotes the $N \times n_s$ regressor matrix

$$\Phi(\xi_s(t)) =$$

$$\begin{bmatrix} \int_{I_1(t)} \delta(\xi - \xi_1(t)) \phi_1(\xi) d\xi & \dots & \int_{I_{n_s}(t)} \delta(\xi - \xi_{n_s}(t)) \phi_1(\xi) d\xi \\ \vdots & \ddots & \vdots \\ \int_{I_1(t)} \delta(\xi - \xi_1(t)) \phi_N(\xi) d\xi & \dots & \int_{I_{n_s}(t)} \delta(\xi - \xi_{n_s}(t)) \phi_N(\xi) d\xi \end{bmatrix},$$

and which simplifies to

$$\Phi(\xi_s(t)) = \begin{bmatrix} \phi_1(\xi_1(t)) & \dots & \phi_1(\xi_{n_s}(t)) \\ \vdots & \ddots & \vdots \\ \phi_N(\xi_1(t)) & \dots & \phi_N(\xi_{n_s}(t)) \end{bmatrix}.$$

For each $t \in [0, T]$, the solution to (17) is given by the least-squares solution

$$\Gamma(t) = (\Phi^T(\xi_s(t))\Phi(\xi_s(t)))^{-1} \Phi^T(\xi_s(t)) (M\kappa(t)). \quad (18)$$

3.2 Sensor network density propagation

Here we associate the kernel $k(t, \xi)$ with the sensor density. The assumption made is that there is a continuum of mobile sensors whose density $\rho(t, \xi)$ is governed by the advection PDE

$$\frac{\partial \rho}{\partial t}(t, \xi) + \frac{\partial}{\partial \xi} (v(t, \xi) \rho(t, \xi)) = 0, \quad (19)$$

having boundary (Dirichlet) and initial conditions

$$\rho(t, 0) = \rho(t, \ell), \quad \rho(0, \xi) = \rho_0(\xi) = 0. \quad (20)$$

The density advection PDE can be formulated on a Hilbert space with the norm describing the energy norm. The modification needed in this formulation is that it must exclude the non-zero constant functions that have zero derivative and this one considers the quotient space, Taylor and Lay (1980) $\mathcal{H} = \bar{L}^2(0, \ell)$. Here $\bar{L}^2(0, \ell)$ is the quotient space of $L^2(0, \ell)$.

The unknown in (19) is the velocity $v(t, \xi)$. Once the density is set equal to the feedback kernel, modulo a constant, then

$$\frac{\partial k}{\partial t}(t, \xi) + \frac{\partial}{\partial \xi} (v(t, \xi) k(t, \xi)) = 0, \quad (21)$$

with $k(t, 0) = k(t, \ell) = 0$. To obtain the unknown sensor network velocity $v(t, \xi)$, one must solve (21) numerically, using approximating functions belonging to a finite dimensional subspace of $\bar{L}^2(0, \ell)$. As an example of the approximating functions for the kernel in (21), one may select Legendre polynomials, Gottlieb and Orszag (1977), shifted for $\Omega = (0, \ell)$ and modified to account for the constants. Viewed in weak form, (21) produces

$$\int_0^\ell \frac{\partial k(t, \xi)}{\partial t} \phi(\xi) d\xi = \int_0^\ell v(t, \xi) k(t, \xi) \frac{d}{d\xi} \phi(\xi) d\xi. \quad (22)$$

The finite dimensional representation of (22) provides the finite dimensional representation of the unknown velocity $v(t, \xi)$.

Once the network velocity is found, one can employ the domain decomposition method used in Section 3.1 to find the speed of a sensor in each cell $I_i(t)$ using the approximation

$$\int_{I_i(t)} v(t, \xi) d\xi \approx v_i(t) \int_{I_i(t)} 1 d\xi.$$

Therefore, the location of each sensor in a cell $I_i(t)$ is implicitly found by defining its initial position $\xi_i(0)$ using the approach in Section 3.1 only once (at the initial time) and then defining its trajectory via its speed $\dot{\xi}_i(t) = v_i(t)$ given by

$$v_i(t) = \frac{\int_{I_i(t)} v(t, \xi) d\xi}{\int_{I_i(t)} 1 d\xi}, \quad i = 1, \dots, n_s. \quad (23)$$

The corresponding static gains are found, in a way similar to the one given in Demetriou (2021), via

$$\gamma_i(t) = \frac{\int_{I_i(t)} \rho(t, \xi) d\xi}{\int_{I_i(t)} 1 d\xi} = \frac{\int_{I_i(t)} k(t, \xi) d\xi}{\int_{I_i(t)} 1 d\xi}, \quad i = 1, \dots, n_s. \quad (24)$$

3.3 Bilinear tracking control

Assume that all sensors are moving within the spatial domain and will not move across the boundary. Here we set

$$v(0) = v(\ell), \quad (25)$$

which makes ξ stay in $\bar{\Omega} = [0, \ell]$ for all t , DiPerna and Lions (1989). Enforcing the density to track the feedback kernel is a natural way to ensure that (25) is justified. In fact, it is easy to check that in this case

$$\begin{aligned} \frac{\partial}{\partial t} \int_0^\ell \rho(t, \xi) d\xi + \int_0^\ell \frac{\partial}{\partial \xi} (v\rho) d\xi = \\ \frac{\partial}{\partial t} \int_0^\ell \rho(t, \xi) d\xi + (v(\ell)\rho(\ell) - v(0)\rho(0)) = 0, \end{aligned}$$

which follows

$$\int_0^\ell \rho(t, \xi) d\xi = \int_0^\ell \rho_0(\xi) d\xi.$$

This guarantees that the number of sensors is conserved.

To find the solution $\rho(t, \xi)$ to (19), one may design the tracking control signal v in an admissible velocity space $U_{ad} = H_0^1(0, T; H_0^1(0, \ell))$ that minimizes

$$\begin{aligned} J(v, \rho) = \frac{1}{2} \|\rho(T) - k(T)\|_{L^2}^2 + \frac{\alpha}{2} \int_0^T \|\rho - k\|_{L^2}^2 dt \\ + \frac{\beta}{2} \int_0^T \left\| \frac{\partial v}{\partial \xi} \right\|_{L^2}^2 dt + \frac{\gamma}{2} \int_0^T \left\| \frac{\partial v}{\partial t} \right\|_{L^2}^2 dt, \quad (P) \end{aligned}$$

subject to (19), where $\alpha, \beta, \gamma > 0$ are the weight parameters. The second and third terms are used to control the acceleration of the sensor movement in space-time as to have a smooth trajectory. Due to the advection term $\nabla \cdot (v\rho)$, the control-to-state map $v \mapsto \rho$ is bilinear. As a result, problem (P) becomes nonconvex. The proofs for the existence of an optimal solution (v^*, ρ^*) and the differentiability of the control-to-state map follow the similar procedures as in Barbu and Marinocchi (2016); Hu (2020). However, the optimal solution may not be unique in general.

To derive the first order necessary optimality conditions for solving problem (P), we shall use a formal Euler-Lagrange approach. The rigorous arguments can be found in Barbu and Marinocchi (2016); Hu (2020). First, introduce the Lagrangian

$$L(\phi, v, \rho) = J(v, \rho) + \int_0^T \left(\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial \xi} (v\rho) \right), \phi dt$$

for $\phi \in H_0^1(0, \ell)$. Applying integration by parts and Green's formula to the second term of L follows

$$\begin{aligned} \int_0^T \left(\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial \xi} (v\rho) \right), \phi dt &= (\rho(T), \phi(T)) - (\rho(0), \phi(0)) \\ &\quad - \int_0^T \left(\frac{\partial \phi}{\partial t}, \rho \right) dt + \int_0^T (v(\ell)\rho(\ell), \phi(\ell)) - (v(0)\rho(0), \phi(0)) dt \\ &\quad - \int_0^T \int_0^\ell (v\rho) \frac{\partial \phi}{\partial \xi} d\xi dt \\ &= (\rho(T), \phi(T)) - (\rho(0), \phi(0)) \\ &\quad - \int_0^T \left(\frac{\partial \phi}{\partial t}, \rho \right) dt - \int_0^T \int_0^\ell (v\rho) \frac{\partial \phi}{\partial \xi} d\xi dt. \end{aligned}$$

Thus

$$\begin{aligned} L(\phi, v, \rho) &= J(v, \rho) + (\rho(T), \phi(T)) + (\rho(0), \phi(0)) \\ &\quad - \int_0^T \left(\frac{\partial \phi}{\partial t}, \rho \right) dt - \int_0^T \int_0^\ell (v\rho) \frac{\partial \phi}{\partial \xi} d\xi dt. \end{aligned}$$

Now setting $\frac{\partial L}{\partial \rho} = 0$ and $\frac{\partial L}{\partial p(T)} = 0$, respectively, we get

$$-\frac{\partial \phi}{\partial t} - \frac{\partial (v\phi)}{\partial \xi} = \alpha(\rho - k), \quad (26)$$

$$\phi(x, T) = -(\rho(x, T) - k(x, T)). \quad (27)$$

Moreover, setting $\frac{\partial L}{\partial v} = 0$ we obtain the optimality conditions

$$\beta \frac{\partial^2 v}{\partial \xi^2} + \gamma \frac{\partial^2 v}{\partial t^2} = \rho \frac{\partial \phi}{\partial \xi}, \quad (28)$$

with boundary and initial conditions

$$v(t, 0) = v(t, \ell) = 0, \quad v(0, \xi) = v(T, \xi) = 0. \quad (29)$$

In summary, to solve problem (P) one has to solve the state equations (19)–(20) and (25) forward in time, coupled with the adjoint system (26)–(27) backward in time together with nonlinear optimality conditions, which are governed by the wave equations (28)–(29).

4. NUMERICAL RESULTS

We consider the following advection-diffusion PDE

$$\begin{aligned} x_t(t, \xi) &= (a(\xi)x_\xi(t, \xi))_\xi + a_2 x_\xi(t, \xi) \\ &\quad + a_3 x(t, \xi) + b(t, \xi)u(t) \end{aligned}$$

where $[0, \ell] = [0, 1]$,

$$a(\xi) = 5 \times 10^{-3} \left(\frac{e^{-g/2}}{\sqrt{2\pi\sigma}} + 1 + 3 \sin(3\pi\xi) \sin^2(\xi - \frac{1}{4}) \right),$$

$\sigma = \ell/18$, $g = (\frac{\xi - \mu}{\sigma})^2$, $\mu = 0.75\ell$, $a_2 = -5 \times 10^{-2}$, $a_3 = -3 \times 10^{-3}$. A Galerkin scheme with 40 linear elements was used to semi-discretize the above PDE and the resulting finite dimensional system was integrated using the 4th order Runge-Kutta stiff ODE solver ode23s from the Matlab[®] ODE library.

The mobile actuator distribution $b(t, \xi)$ was selected as the boxcar function $b(t, \xi) = 1$ if $\xi \in [\xi_a(t) - \frac{\varepsilon}{2}, \xi_a(t) + \frac{\varepsilon}{2}]$, $\varepsilon = \ell/20$, and zero otherwise, having a centroid with path

$$\xi_a(t) = 0.5 - 0.45 \sin(4\pi(t_f - t_0)/(t - t_0)),$$

and the initial condition was selected as $x(0, \xi) = 10 \sin(\pi(1 - \xi))e^{-7(\xi - 1)^2}$. The solution to (4) used $Q = I$ and $R = 10^{-2}$.

Figure 2 depicts the spatial distribution of the state at four different time instances using the full state feedback controller (12) and the approximated static output feedback controller

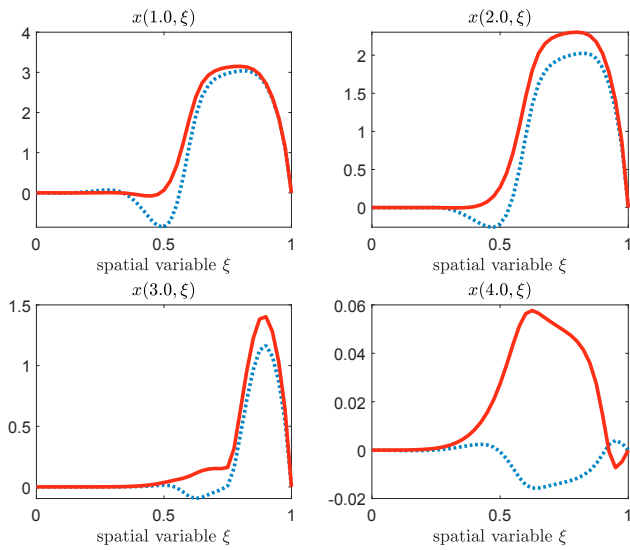


Fig. 2. Spatial evolution of state at four time instances; full state controller (blue dotted), output feedback (red solid).

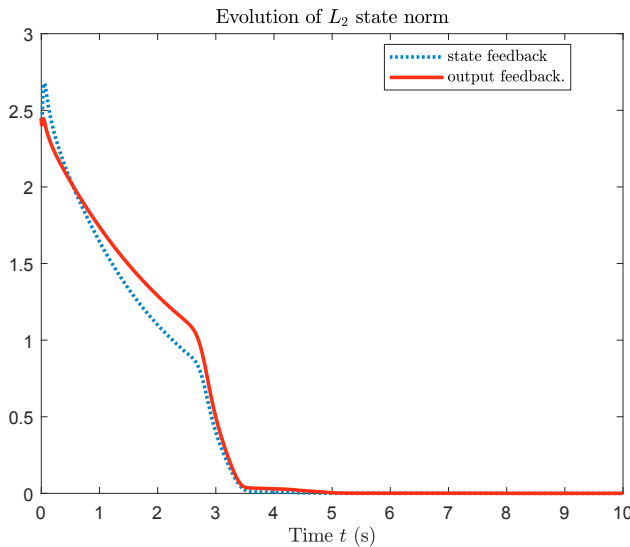


Fig. 3. Evolution of state L_2 norm.

(13). The performance of the state feedback controller cannot be surpassed by the static output feedback controller, but such a controller is realizable and has a performance that is close to that of the state feedback. The same is observed in Figure 3 which depicts the evolution of the state L_2 norm. The path evolution of the $n_s = 12$ mobile sensors is depicted in Figure 4, where it is observed that the proposed μ -CVT method ensures collision avoidance and all sensors remain within Ω .

5. CONCLUSION AND FUTURE WORK

A way to incorporate mobile sensor networks for the control of PDEs is to relate the density of a sensor network to the kernel associated with a feedback operator. When the state feedback cannot be implemented, then a sensor network can be used as an alternate to an observer-based compensator. This takes the form of a static output feedback and the sensor network-based controller provides the static gains and the sensor positions as a means to approximate a full state feedback. Three different approaches were presented to approximate a state feedback

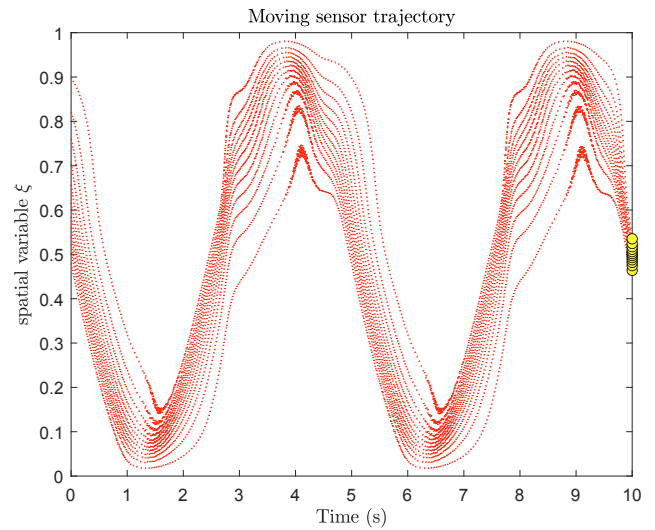


Fig. 4. Evolution of the sensor network trajectories.

controller by a static output feedback controller that utilized the measurements obtained by a mobile sensor network. An immediate extension considered by the authors is the application to 3D PDEs and the inclusion of the motion dynamics of the mobile platforms carrying the sensors.

REFERENCES

- Barbu, V. and Marinocchi, G. (2016). An optimal control approach to the optical flow problem. *Systems & Control Letters*, 87, 1–9.
- Curtain, R.F. and Zwart, H.J. (1995). *An Introduction to Infinite Dimensional Linear Systems Theory*. Springer-Verlag, Berlin.
- Demetriou, M.A. (2021). Employing mobile sensor density to approximate state feedback kernels in static output feedback control of pdes. In *Proc. of the American Control Conference*, 2775–2781. doi:10.23919/ACC50511.2021.9483372.
- DiPerna, R.J. and Lions, P.L. (1989). Ordinary differential equations, transport theory and sobolev spaces. *Inventiones mathematicae*, 98(3), 511–547.
- Gottlieb, D. and Orszag, S.A. (1977). *Numerical analysis of spectral methods: theory and applications*. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, Philadelphia, PA.
- Hu, W. (2020). An approximating control design for optimal mixing by stokes flows. *Applied Mathematics & Optimization*, 82(2), 471–498.
- King, B.B. (1995). Existence of functional gains for parabolic control systems. In *Computation and control, IV (Bozeman, MT, 1994)*, volume 20 of *Progr. Systems Control Theory*, 203–217. Birkhäuser Boston, Boston, MA.
- King, B.B. (2000). Representation of feedback operators for parabolic control problems. *Proc. Amer. Math. Soc.*, 128(5), 1339–1346. doi:10.1090/S0002-9939-00-05647-1.
- Pisano, A. and Orlov, Y. (2017). On the ISS properties of a class of parabolic DPS' with discontinuous control using sampled-in-space sensing and actuation. *Automatica J. IFAC*, 81, 447–454. doi:10.1016/j.automatica.2017.04.025.
- Showalter, R.E. (1977). *Hilbert Space Methods for Partial Differential Equations*. Pitman, London.
- Taylor, A.E. and Lay, D.C. (1980). *Introduction to functional analysis*. John Wiley & Sons, New York-Chichester-Brisbane, second edition.