

Employing mobile sensor density to approximate state feedback kernels in static output feedback control of PDEs

Michael A. Demetriou

Abstract—This work considers the replacement of a full-state feedback controller by a static output feedback controller employing a finite number of point sensors. This is achieved by the approximation of the feedback kernel associated with the full state feedback operator. The feedback kernel is partitioned into equiareal cells and an appropriately selected centroid within each cell serves as the sensor location. This allows one to approximate the inner product of the feedback kernel and the full state by the finite weighted sum of static output feedback measurements. By equating the feedback kernel with the density of a hypothetical sensor network, the problem of approximating the sensor density becomes that of partitioning the sensor density using the proposed computational-geometry based decomposition that is based on a modification of Centroidal Voronoi Tessellations. When the control is considered over a finite horizon and/or the actuator itself is repositioned within the spatial domain, the resulting feedback kernel is rendered time-varying. This requires its partitioning at each time leading to mobile sensors within the spatial domain. Two guidance policies are proposed: one uses the partitioning of the kernel method at each time to find the optimal sensors thus resulting in moving sensors. The other method uses the kernel partitioning only at the initial time and subsequently uses the sensor density as the initial condition for an advection PDE that represents the evolution of the sensor density. This advection PDE is solved for the velocity thereby providing the velocity of the density of the sensor network. Projecting the sensor density velocity onto the same partitioning used for the kernel provides the sensor velocities. A numerical example of an advection diffusion PDE is presented to provide an understanding of this computational geometry based partitioning of feedback kernels.

I. INTRODUCTION

This paper examines an alternate to the implementation of an observer-based feedback for the control of distributed parameter systems. For a class of these systems, written as evolution equations in a functional space, the state feedback operator admits a kernel representation, [1]. The control signal is then equal to the inner product of the feedback kernel and the infinite dimensional system. This inner product representation was used as a means to select sensor locations using the following argumentation: since the feedback kernel serves as a spatial weight in the inner product, then in the spatial regions where the kernel is “larger”, it designates a larger importance of the state and hence it provides a good candidate for sensor placement. In fact this was explored in a series of works in [2], [3], [4], [5].

Another effort exploring the use of the functional gain (kernel) to help with the control design was examined in

[6] where the kernel was partitioned into equiareal cells using a modification of the Centroidal Voronoi Tessellations (CVT) method, in which a sensor was placed in each cell. By approximating the full state feedback controller, expressed as the inner product of the kernel and the state, using static output feedback, one opts to avoid the use of computationally-intensive state estimators. This approximation aims at replacing the inner product of the kernel and the state by a weighted sum of the gains and the state measurements, by minimizing the error between the inner product (kernel and state) and the weighted sum of sensor measurements. The earlier work [7], [8] explore this inner product representation but used ad-hoc methods for sensor placement.

In the event that a mobile actuator is implemented, or the state operator is time-dependent, then the feedback kernel is also time varying. In order to approximate this inner product by a weighted sum of sensor measurements at each time, one must allow the sensors to move throughout the spatial domain. Thus one arrives at time-varying static gains and moving sensors. At this stage, one is faced with the following tasks: how to compute the gains and how to repositioned the mobile sensors in the spatial domain, the best approximate the inner produce of the kernel and the state.

This paper addresses the above questions by proposing two different control approximation strategies. In the first one, the modified CVT are applied at each time to partition the kernel into n_s equiareal cells and place a sensor in each. Then using the volume of the cell under the feedback kernel, it computes the corresponding static gain. This is implemented in each time, thereby rendering both the feedback gains and the sensor locations time varying. The other approach uses the first method only for the initial time and uses the gas-kinetic-based model first used in a series of works in [9], [10], [11], [12] to represent the continuum of the agents (sensors) for a macroscopic model in terms of the advection PDE governing the sensor density. Once the initial deployment of the sensors is obtained, then the kernel can be used as the approximation of the sensor density and thus the advection PDE for the density is used to propagate the mobile sensors via the solution of the macroscopic velocity. The associated time-varying static gains are computed in the same manner as the first method utilizing the feedback kernel.

Therefore the contribution of thus work is summarized as

- 1) Present a control approximation scheme whereby an idealized full state feedback controller is approximated by the weighted sum of pointwise process measurements resulting in a static output feedback controller.
- 2) Propose a modification of the Centroidal Voronoi

M. A. Demetriou is with WPI, Aerospace Engineering Department, Worcester, MA 01609, USA, mdemetri@wpi.edu. The author acknowledges financial support from NSF-CMMI grant # 1825546.

Tessellation method to partition the feedback kernel into n_s equiareal cells and subsequently place a single pointwise sensor in each of these cells

- 3) Compute the associated static gains for each time that best approximate a full-state feedback controller.
- 4) Propose two guidance schemes for the spatial repositioning of the mobile sensors. One method applies the above kernel decomposition and gain estimation at each time. The other guidance uses only the initial positioning of the sensors computed from the kernel approximation at the initial time and uses the sensor locations as the approximation of the sensor density corresponding to a network of infinite number of pointwise sensors at the initial time. The sensor velocities are computed from the solution to the advection PDE that describes the sensor density where the density is substituted by the feedback kernel.

The problem of full state feedback control is presented in Section II. The proposed method for the kernel partitioning and the associated selection of static gains is demonstrated in Section III. For the time-dependent kernel, the kernel partitioning is applied in each time and the two sensor guidances are presented Section IV. Simulation results are presented in Section V and conclusions follow in Section VI.

II. PROBLEM FORMULATION

The spatially distributed processes under consideration are represented by evolution equations in a Hilbert space $\mathcal{X}$

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

where the state operator $\mathcal{A} \in L(\mathcal{V}, \mathcal{V}^*)$ and $\mathcal{B} \in L(\mathcal{U}, \mathcal{V}^*)$ is the input (control) operator. The control space is denoted by the finite dimensional Euclidean space $\mathcal{U}$. Since the input and output operators are defined in different spaces, we define the state space $\mathcal{X}$, which is a Hilbert space as the interpolating space and the reflexive Banach space $\mathcal{V}$ that is continuously and densely embedded in $\mathcal{X}$. The conjugate dual of $\mathcal{V}$ is denoted $\mathcal{V}^*$. In this case, one has that $\mathcal{V} \hookrightarrow \mathcal{X} \hookrightarrow \mathcal{V}^*$ with both embeddings dense and continuous.

It is assumed that in the ideal case, the full state is available. Furthermore, either the state and input operators are time-varying and/or a finite horizon LQR problem is considered. For the simpler case, it is assumed that the input operator is rendered time-varying via the spatial repositioning of the actuator. The specifics of this actuator motion are important only to the extent that they result in a time-varying feedback gain. The derivation of the actuator guidance and the computation of the associated full state feedback gain, that is time-varying, are not considered in this paper.

For example, when a time-varying input operator $\mathcal{B}(t)$ is assumed, then an LQR-based controller is derived by minimizing the finite horizon linear quadratic functional

$$J = \int_0^T \langle x(\tau), Qx(\tau) \rangle_{\mathcal{X}} + u^T(\tau) R^{-1} u(\tau) d\tau + \langle x(T), \mathcal{M}x(T) \rangle \quad (2)$$

subject to (1) with $\mathcal{B}$ substituted by its time varying version

$\mathcal{B}(t)$ The full state feedback control law is given by [13]

$$u_f(t) = -\mathcal{K}(t)x(t). \quad (3)$$

The time-varying feedback gain operator $\mathcal{K}(t) : \mathcal{V} \rightarrow \mathcal{U}$ is derived from the Operator Differential Riccati Equation

$$-\dot{\mathcal{P}} = \mathcal{A}^* \mathcal{P} + \mathcal{P} \mathcal{A} - \mathcal{P} \mathcal{B}(t) R^{-1} \mathcal{B}^*(t) \mathcal{P} + Q = 0, \quad (4)$$

where the input operator is explicitly dependent on time in order to emphasize that in systems with moving actuators it is time varying. Equation (4) is supplemented with the terminal condition $\mathcal{P}(T) = \mathcal{M}$. The feedback operator is given by

$$\mathcal{K}(t) = R^{-1} \mathcal{B}^*(t) \mathcal{P}(t). \quad (5)$$

As the full-state feedback (3) cannot be implemented, one must seek alternative ways to realize it. The one adopted here, essentially approximates the control (3) by a finite sum of weighted measurements

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

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

To better understand the approximation of $u_f(t)$ in (3) by $u_s(t)$ in (6), we describe the sensor measurements given by

$$y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_{n_s}(t) \end{bmatrix} = \begin{bmatrix} C_1(t)x(t) \\ \vdots \\ C_{n_s}(t)x(t) \end{bmatrix} = C(t)x(t). \quad (7)$$

Therefore, the control approximation becomes

$$(\mathcal{K}(t) - \Gamma C(t))x(t) \approx 0 \quad (8)$$

The above is viewed in weak form in the appropriate space. To improve the above approximation, one also allows the sensor locations to be selected at each time in order to minimize, in the appropriate sense, the above difference.

In order to use the proposed computational geometry method, it is assumed that the feedback gain $\mathcal{K}(t)$ admits a kernel representation. To better appreciate this, we consider the 1D advection PDE, which falls under the proposed class of systems (1). Thus, we have

$$\frac{\partial x}{\partial t}(t, \xi) = \frac{\partial}{\partial \xi} \left(a(\xi) \frac{\partial x}{\partial \xi}(t, \xi) \right) + b(\xi, t) u(t) \\ y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_{n_s}(t) \end{bmatrix} = \begin{bmatrix} \int_0^\ell c_1(\xi, \xi_1(t)) x(t, \xi) d\xi \\ \vdots \\ \int_0^\ell c_{n_s}(\xi, \xi_{n_s}(t)) x(t, \xi) d\xi \end{bmatrix}, \quad (9)$$

with boundary conditions $x(t, 0) = x(t, \ell) = 0$ and initial condition $x(0, \xi) = x_0(\xi)$. The parameter $a(\xi)$ is the thermal diffusivity and $b(\xi, t)$ denotes the spatial distribution of the mobile actuator. Since we also allow the sensors to move, they are parameterized by their time-varying centroid (spatial location) and thus their spatial distribution is denoted by $c_i(\xi, \xi_i(t))$, $i = 1, \dots, n_s$, where $\xi_i(t)$, $i = 1, \dots, n_s$ denote their time-varying centroids. In relation to the spaces associated with (1), the state space is $\mathcal{X} = L_2(0, \ell)$ with $\mathcal{V} = H_0^1(0, \ell)$.

Similar to the earlier work in [6], we assume that the sensing devices are identical, and that they differ in their spatial location, as dictated by their centroids $\xi_i(t)$. Thus, we adopt the uniform notation

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

The control law (3) for the PDE in (8) takes the form

$$u_f(t) = - \int_0^\ell k(t, \xi) x(t, \xi) d\xi. \quad (10)$$

The approximation (8) now takes the form

$$\int_0^\ell k(t, \xi) x(t, \xi) d\xi \approx \sum_{i=1}^{n_s} \gamma_i(t) y_i(t). \quad (11)$$

Involving the spatial distribution of the sensors, it becomes

$$\int_0^\ell k(t, \xi) x(t, \xi) d\xi \approx \sum_{i=1}^{n_s} \gamma_i(t) \int_0^\ell c(\xi, \xi_i(t)) x(t, \xi) d\xi. \quad (12)$$

This of course, is enforced in weak form and for a test function $\phi \in H_0^1(0, \ell)$ we have to minimize the Euclidean norm of the difference of the control signals

$$\min_{\phi \in H_0^1(0, \ell)} \left| \int_0^\ell \left(k(t, \xi) - \sum_{i=1}^{n_s} \gamma_i(t) c(\xi, \xi_i(t)) \right) \phi(\xi) d\xi \right|^2 \quad (13)$$

The optimization (13) will produce, for each time $t \in [0, T]$, the optimal sensor locations ξ_i and the associated static feedback gain γ_i , $i = 1, \dots, n_s$.

The question that arises from the optimization (13) is how to best approximate the feedback kernel $k(t, \xi)$ in each time t by the static output feedback in (12). The idea considered here is to partition the kernel $k(t, \xi)$ with respect to the spatial domain Ω so that each partition will carry the same level of control authority. In each of those n_s partitions, a single sensor will be placed via the sensor centroid ξ_i and one must only find what is the corresponding static gain γ_i .

The kernel-based partitioning will form the first part of the above optimization. Once this is achieved and a single sensor is placed within each of these partitions, the second part of the optimization will yield the static gains.

III. KERNEL-BASED PARTITIONING

Here, a simple example of a constant control signal, which is formed as the $L_2(0, \ell)$ inner product of the kernel $k(\xi)$ and state $x(\xi)$ is considered. The kernel and state are selected as

$$k(\xi) = 2e^{-10(\xi-0.1)^2} \sin(\pi\xi), \quad x(\xi) = 2e^{-10(\xi-0.9)^2} \sin(\pi\xi),$$

and are depicted in Figure 1. One is interested in evaluating their inner product

$$\int_0^\ell k(\xi) x(\xi) d\xi.$$

In particular, one is interested in approximating this integral

$$\int_0^\ell k(\xi) x(\xi) d\xi \approx \sum_{i=1}^{n_s} \gamma_i \int_0^\ell \delta(\xi - \xi_i) x(\xi) d\xi = \sum_{i=1}^{n_s} \gamma_i x(\xi_i). \quad (14)$$

This is interpreted as approximating the inner product by the weighted sum of the state evaluated at the pointwise sensor locations; here the sensors have a spatial distribution equal to the Dirac delta function centered at the sensor

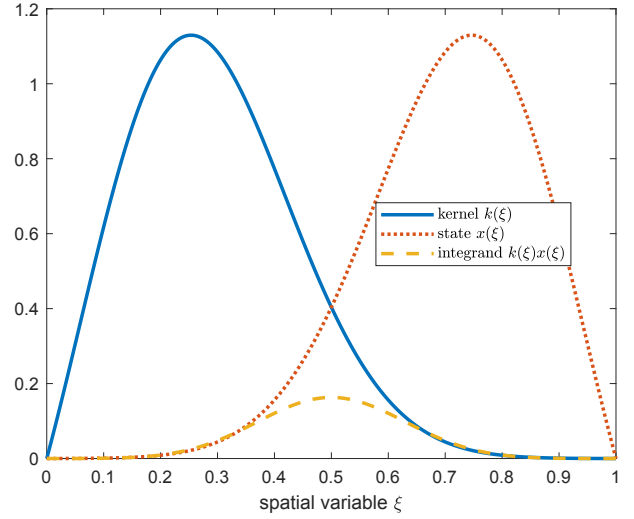


Fig. 1. Kernel $k(\xi)$, state $x(\xi)$ and the integrand $k(\xi)x(\xi)$ in the inner product $\langle k, x \rangle = \int_0^\ell k(\xi)x(\xi) d\xi$. The area under the integrand $k(\xi)x(\xi)$ is the value of the integral $\langle k, x \rangle$.

centroids ξ_i . Once the sensor centroids are computed (using the proposed modified CVT partitioning) then the weights γ_i can be obtained via a couple of methods presented here.

The product $k(\xi)x(\xi)$ (orange line) in Figure 1 is the integrand and whose area will yield the value of the inner product. This area will be approximated by the pointwise values of $x(\xi)$ and some weights that are related to the partitioning of $k(\xi)$. As the kernel carries the information where the state is “more” important, then it also serves as the density of a continuum of sensors in the expansion

$$\int_0^\ell k(\xi)x(\xi) d\xi = \sum_{i=1}^{\infty} k(\xi_i)x(\xi_i).$$

It is easily seen above that in the spatial regions where $k(\xi_i)$ is larger, then the weight in the expansion should be higher. Therefore, sensors should be placed in spatial regions where the value of the kernel is larger. This forms the theoretical cornerstone of the work undertaken in this paper.

A. Kernel partitioning and sensor position selection

Here, the spatial domain $\Omega = [0, 1]$ is partitioned into n_s cells that have the same area under the kernel. The CVT method is modified as follows. This procedure is similar to the one presented in [6] and is summarized below.

- 1) Partition the spatial domain Ω in n_s cells $I_i \in \bar{\Omega}$ with $I_i \cap I_j = \emptyset$ and $\bigcup_{i=1}^{n_s} I_i = \bar{\Omega}$ so that at each cell I_i , the area of the kernel $k(\xi)$ satisfies

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

- 2) Place a sensor ξ_i in each cell I_i , using again the proposed method in step #1 which ensures that the sensor location ξ_i in cell I_i is such that it subsequently divides the cell into two subcells $I_i = I_i^a \cup I_i^b$ of equal areas of the kernel, with each being equal to $0.5A/n_s$.

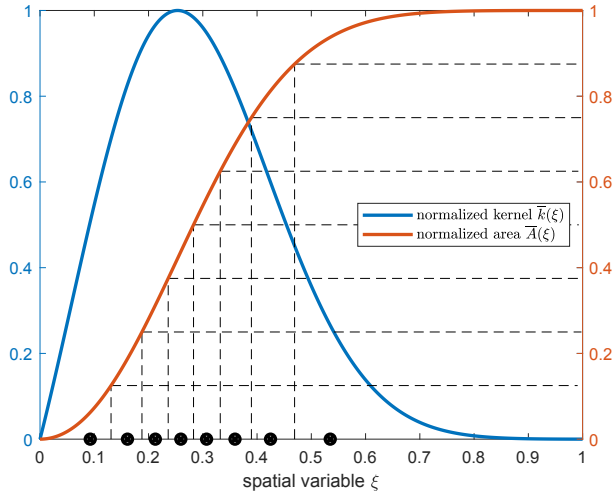


Fig. 2. Normalized kernel $\bar{k}(\xi)$ and normalized areal function $\bar{A}(\xi)$. The domain Ω is partitioned (grid generation) into $n_s = 8$ equiareal cells I_i with $\int_{I_i} k(\xi) d\xi = \frac{1}{n_s} \int_0^1 k(\xi) d\xi$. The cell coordinates (grid points) are found as the abscissae points on the normalized areal function $\bar{A}(\xi)$ having ordinates 0.125, 0.250, 0.375, 0.500, 0.625, 0.770, 0.875 given by i/n_s , $i = 1, \dots, n_s$.

For $i = 1, \dots, n_s$, the sensor location are found via

$$\xi_i : \int_{I_i^a(\xi_i)} k(\xi) d\xi = \int_{I_i^b(\xi_i)} k(\xi) d\xi = \frac{1}{2} \int_{I_i} k(\xi) d\xi. \quad (16)$$

For the 1D case, there is a graphical method that enables one to easily compute the sensor locations. First, define the normalized kernel $\bar{k}(\xi)$ and the normalized area function under the kernel as follows

$$\bar{k}(\xi) = \frac{k(\xi)}{\max_{\xi \in \Omega} k(\xi)}, \quad \bar{A}(\xi) = \frac{A(\xi)}{A(\ell)}, \quad A(\xi) = \int_0^\xi k(\xi) d\xi. \quad (17)$$

Please note that $A(\ell) = A$ is the total area under the kernel and thus $\bar{A}(\xi)$ represents the fraction of the area under the kernel up to point ξ ; it is easily seen that $\bar{A}(\ell) = 1$.

To locate the cells (intervals) I_i using the area function $\bar{A}(\xi)$, all is needed it so identify the ordinate points i/n_s of $\bar{A}(\xi)$. The corresponding abscissae points in the graph of $\bar{A}(\xi)$ will immediately yield the spatial points that partition the spatial domain Ω into n_s equiareal cells. It turns out that one does not really need to find the grid points in the 1D case. The reason is that each equiareal cell must subsequently be partitioned into two subcells of equal areas. These points that partition each cell are found via

$$\xi_i : \int_0^{\xi_i} k(\xi) d\xi = \frac{(2i-1)}{2} \frac{A}{n_s}, \quad i = 1, \dots, n_s. \quad (18)$$

Figure 2 depicts the normalized kernel $\bar{k}(\xi)$ and the normalized areal function $\bar{A}(\xi)$. The $\bar{A}(\xi)$ axis (red) is decomposed, via a uniform grid, into $n_s = 8$ cells with the grid points given by the ordinates i/n_s , $i = 1, \dots, n_s$. The corresponding abscissae points immediately give the grid points on the spatial variable axis that decompose the normalized kernel $\bar{k}(\xi)$ into n_s cells of equal area.

It should be noted that the grid on the ξ axis is not uniform since they decompose the area function $k(\xi)$ into cells (intervals) of equal area. However the strength of this approach is that the vertical axis (the $\bar{A}(\xi)$ -axis is

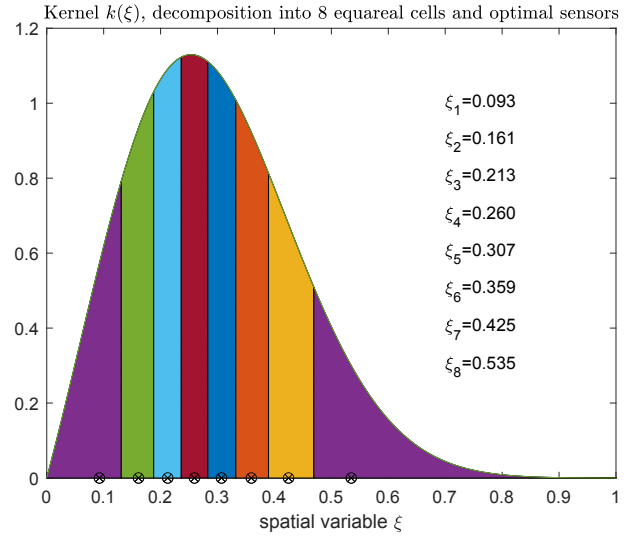


Fig. 3. Kernel $k(\xi)$ and the associated equiareal cells with the corresponding sensor locations.

divided into n_s cells using a uniform grid. Simply put, *the normalized areal function $\bar{A}(\xi)$ axis uses a uniform grid and the corresponding abscissae points provide the non-uniform grid points where the normalized kernel $\bar{k}(\xi)$ is decomposed into cells of equal areas!*

The sensor locations and the corresponding cells are depicted in Figure 3 for the function $k(\xi)$.

B. Kernel-based gains γ_i in (14)

Two different methods are summarized for the calculation of the gains γ_i that best approximate (14). The first one is taken from [6] and is presented first.

The kernel is assumed to admit an expansion

$$k(\xi) = \sum_{j=1}^N k_j \phi_j(\xi), \quad \forall \phi_j \in H_0^1(0, 1). \quad (19)$$

This is replaced in the approximation (14). Since this should be valid for all functions $\phi(\xi)$, then (14) is considered with $x(\xi)$ replaced by a test function $\phi \in H_0^1(0, 1)$. We then have that the finite dimensional representation of (14) with $x(\xi)$ replaced by a test function is

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

Since it is true for all test functions $\phi \in H_0^1(0, 1)$ then it is true for all (trial) functions $\phi_m(\xi)$. When the above is evaluated for all trial functions and with the spatial distribution of the sensors replaced by the Dirac delta functions, one arrives at

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

for all test functions $\phi_m \in H_0^1(0, 1)$, $m = 1, \dots, N$. This in matrix form is written as

$$M\kappa = \Phi(\xi_s)\Gamma \quad (20)$$

where the $N \times N$ mass matrix M is given by

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

The vector of the kernel coefficients κ and the vector of the static gains are given by

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

Finally, the $N \times n_s$ regressor matrix $\Phi(\xi_s)$ which relates to the observability matrix

$$\Phi(\xi_s) = \begin{bmatrix} \int_{I_1} c(\xi_1) \phi_1(\xi) d\xi & \dots & \int_{I_{n_s}} c(\xi_{n_s}) \phi_1(\xi) d\xi \\ \vdots & \ddots & \vdots \\ \int_{I_1} c(\xi_1) \phi_N(\xi) d\xi & \dots & \int_{I_{n_s}} c(\xi_{n_s}) \phi_N(\xi) d\xi \end{bmatrix},$$

which for the case of pointwise sensors simplifies to

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

Since the approximation (20) is not exact, but has an error in the right hand side, then a least squares method is used to solve for Γ ; the $L^2(\mathbb{R}^N)$ norm of the error in (20) is

$$\Gamma = (\Phi^T(\xi_s) \Phi(\xi_s))^{-1} \Phi^T(\xi_s) (M\kappa). \quad (21)$$

Equation (21) requires the coefficients of the approximation to the kernel $k(\xi)$ and the regression matrix that is the finite dimensional representation of the observability matrix. This is computed by evaluating the N test functions at the n_s sensor locations. As part of the solvability of the least squares method, one must require that $\Phi(\xi_s)$ has rank n_s . This can be ensured either by selecting the sensor locations in each cell I_i that would yield the requisite rank condition, or by appropriately selecting the test functions used in the finite dimensional approximation (19).

The second approach to compute the gains γ_i in (14) once the sensor locations are obtained via the proposed modified CVT method, is based on the average height of each cell that produces the same area as each cell. It is given by

$$\gamma_i = \left(\frac{A}{n_s} \right) \frac{1}{\text{meas}(I_i)}, \quad i = 1, \dots, n_s. \quad (22)$$

For the 1D case, this simplifies to

$$\gamma_i = \left(\frac{A}{n_s} \right) \frac{1}{\delta \xi_i} \quad i = 1, \dots, n_s, \quad (23)$$

where $\delta \xi_i$ is the length of the i^{th} interval (cell). Please note that the intervals $\delta \xi_i$ are not uniform, thereby resulting in different gains γ_i .

As a demonstration, consider the two graphs of $k(\xi)$ and $x(\xi)$ depicted in Figure 1 over the interval $[\xi_1, \xi_2]$. Figure 4 depicts the shaded area under the curve $k(\xi)x(\xi)$. Numerical integration yields

$$\int_{\xi_1}^{\xi_2} k(\xi)x(\xi) d\xi \approx k(\xi_{av})x(\xi_{av})(\xi_2 - \xi_1) = x(\xi_{av}) \int_{\xi_1}^{\xi_2} k(\xi) d\xi.$$

With regards to the kernel partitioning, the above area

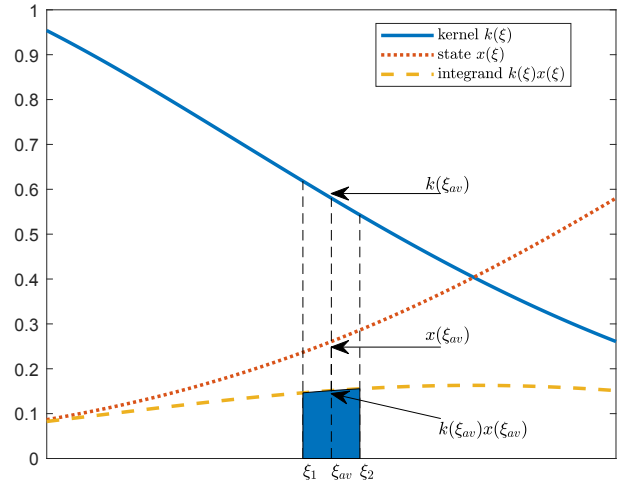


Fig. 4. Numerical approximation of the area under the curve $k(\xi)x(\xi)$ over the interval $[\xi_1, \xi_2]$. The shaded area $k(\xi_{av})x(\xi_{av})(\xi_2 - \xi_1)$ is equivalently approximated by $x(\xi_{av}) \int_{\xi_1}^{\xi_2} k(\xi) d\xi$.

approximation yields the approximation of the area under the curve $k(\xi)x(\xi)$ over a cell I_i as follows

$$\int_{I_i} k(\xi)x(\xi) d\xi \approx x(\xi_i) \int_{I_i} k(\xi) d\xi.$$

This shows that the inner product over a cell is approximated by the value of the state at the point ξ_i (output) multiplied by the area of the kernel over the cell and which in turn shows that the static feedback gains are given by (23). It should be noted that for the uniform grid case, the static gains simplify to $\gamma_i = k(\xi_i)$.

IV. MAIN RESULTS

Once the feedback kernel is partitioned into n_s cells and a pointwise sensor is placed in each cell, then either of the two methods presented in Section III-B can be used to obtain the associated static gains. This can be used to time-varying feedback kernels at each time. Thus (14) becomes

$$\int_0^\ell k(t, \xi)x(t, \xi) d\xi \approx \sum_{i=1}^{n_s} \gamma_i(t) \int_0^\ell c(\xi, \xi_i)x(t, \xi) d\xi.$$

For the case of pointwise in space sensor distributions (i.e. spatial delta functions) it simplifies to

$$\begin{aligned} \int_0^\ell k(t, \xi)x(t, \xi) d\xi &\approx \sum_{i=1}^{n_s} \gamma_i(t) \int_0^\ell \delta(\xi - \xi_i)x(t, \xi) d\xi \\ &= \sum_{i=1}^{n_s} \gamma_i(t)x(t, \xi_i(t)). \end{aligned}$$

The state evaluated at the sensor locations is $x(t, \xi_i(t))$ and represents the output of each mobile sensor since

$$y_i(t) = \int_0^\ell \delta(\xi - \xi_i(t))x(t, \xi) d\xi = x(t, \xi_i(t)).$$

The static gains $\gamma_i(t)$ are dependent on the now time varying kernel and are thus time varying as well.

If at each time, the procedure presented in Section III is applied, then the sensor locations will be changing, thus leading to moving sensors. This is essentially a kernel-partitioning based guidance.

An alternate method is also examined. Assuming a continuum of pointwise sensors is used to approximate $k(t, \xi)$, then the feedback kernel can be thought of as the sensor density. Such a density is governed by the advection PDE

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial \xi} (v\rho) = 0 \quad (24)$$

with Dirichlet boundary conditions $\rho(t, 0) = \rho(t, \ell)$ and an initial sensor density $\rho(0, \xi) = \rho_0(\xi)$.

Ideally, one would like to select $v(t, \xi)$ of the sensor density so that at each time it matches the sensor location and static gains that best approximate the feedback kernel. However, since it is the kernel that is being approximated, then one can use the kernel as the sensor density and instead of solving for the sensor locations at each time, can solve for the sensor velocities. To do so, one must first use the method in Section III to obtain the initial sensor density (initial sensor locations) and then substitute the kernel $k(t, \xi)$ in the above advection PDE in order to obtain the velocity $v(t, \xi)$. Since one does not have an infinite number of sensors, but rather a finite number n_s is used to decompose the kernel, then the same decomposition can be used to decompose the velocity $v(t, \xi)$ in order to find the velocity of each cell and thus the velocity of each sensor. Possible issues with uniqueness of the velocity v in (24) may arise due to the fact that the kernel may be related to the density modulo a constant $k(t, \xi) \sim \rho(t, \xi) + \text{constant}$, thus requiring one to consider a quotient space for (24). Another issue arising is the fact that a separate velocity equation along with an equation for the equilibrium speed variance which accounts for pressure terms, as was presented in [10], is required. For the 1D case, one may ignore the velocity and speed variance equations and only consider (24), but must make additional assumptions to ensure the uniqueness of $v(t, \xi)$.

V. NUMERICAL EXAMPLES

An advection-diffusion PDE, similar to the one in (9)

$$x_t(t, \xi) = (a(\xi)x_\xi(t, \xi))_\xi + a_2 x_\xi(t, \xi) + a_3 x(t, \xi) + b(\xi, t)u(t)$$

was considered for the simulation study. The domain was selected as $\Omega = [0, 1]$ and a spline-based Galerkin approximation scheme was used to produce a semi-discretized system of linear differential equations. Additionally, the approximation scheme selected ensured that exponential stabilizability is preserved [14], [15]. The requisite spatial integrals required for the numerical computation of the matrix representation of the PDE in (9) were computed using a composite two-point Gauss-Legendre quadrature rule [16]. The finite dimensional state space model resulting from the Galerkin approximation was subsequently integrated using the stiff ODE solver from the Matlab[®] ODE library, routine `ode23s`, a 4th order Runge-Kutta scheme. The Galerkin scheme used 80 linear spline elements with the parameters set to

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

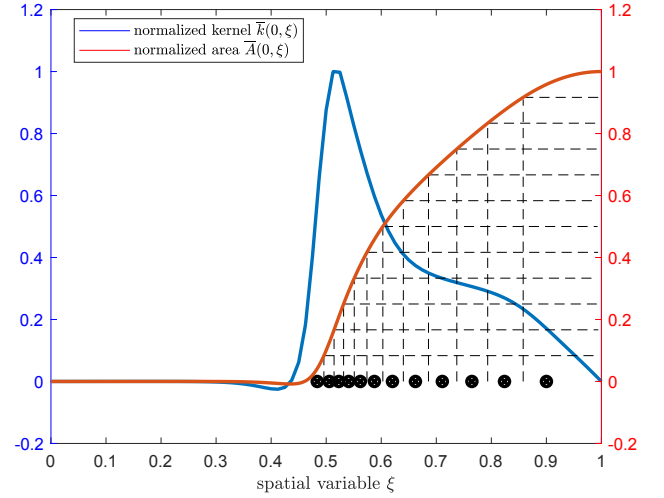


Fig. 5. Initial normalized feedback kernel $\bar{k}(0, \xi)$ and normalized areal function along with the kernel partitioning and associated sensor locations.

$\sigma = \ell/18$, $g = (\frac{\xi - \mu}{\sigma})^2$, $\mu = 0.75\ell$, $a_2 = -5 \times 10^{-2}$, $a_3 = -3 \times 10^{-3}$. The mobile actuator spatial distribution was selected as the boxcar function $b(\xi, \xi_a(t)) = 1$ if $\xi \in [\xi_a(t) - \frac{\varepsilon}{2}, \xi_a(t) + \frac{\varepsilon}{2}]$, $\varepsilon = \ell/20$, and zero otherwise, with a moving centroid whose path was governed by

$$\xi_a(t) = 0.5 \left(1 - 0.9 \sin(2\pi \frac{2(t-t_i)}{t-t_i}) \right)$$

and $x(0, \xi) = 20 \sin(\pi(1 - \xi)) e^{-7(\xi-1)^2}$.

For the solution to the operator Riccati equation (4), the LQR parameters in (2) were selected as $Q = I$ and $R = 10^{-2}$. Finally, the sensor distributions were assumed to be spatial delta functions with $c(\xi_i(t)) = \delta(\xi - \xi_i(t))$, $i = 1, \dots, n_s = 12$.

The optimal full state feedback was found via the solution to the operator Riccati equation over the time interval $[0, 12]$ s. Subsequently, the feedback kernel $k(t, \xi)$ was computed and stored. At the initial time $t_0 = 0$, the procedure summarized in Section III was implemented to find the initial sensor location (and distribution) and the associated static feedback gain.

Figure 5 depicts the initial normalized feedback kernel along with the normalized areal function used in order to obtain the kernel decomposition. The initial feedback kernel was partitioned into $n_s = 12$ equiareal cells. The sensor distribution corresponding to the sensor placement at the initial time $t_0 = 0$ is presented in Figure 6.

The temporal repositioning of the n_s sensors was implemented with the successive application of the kernel partitioning described in Section III-A, and the derivation of the time varying gains was realized via the successive application of (23).

While the approximation of the full state feedback by static output feedback cannot surpass the performance of the full state feedback controller as demonstrated in Figure 7, it nonetheless provide an inexpensive alternative. Indeed, Table I tabulates the $L_2(0, T)$ norm for the state norm and the control effort. it is observed that for a comparable performance, the static output feedback that utilized a mobile sensor network results in a significantly lower control effort.

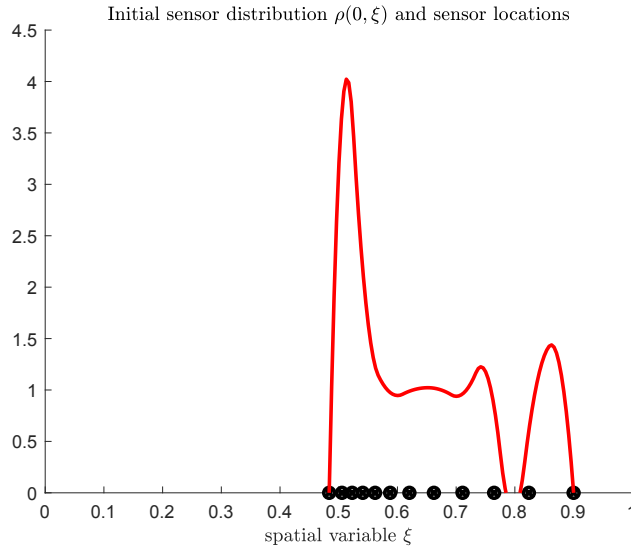


Fig. 6. Sensor locations for initial time and corresponding sensor distribution $\rho(0, \xi)$.

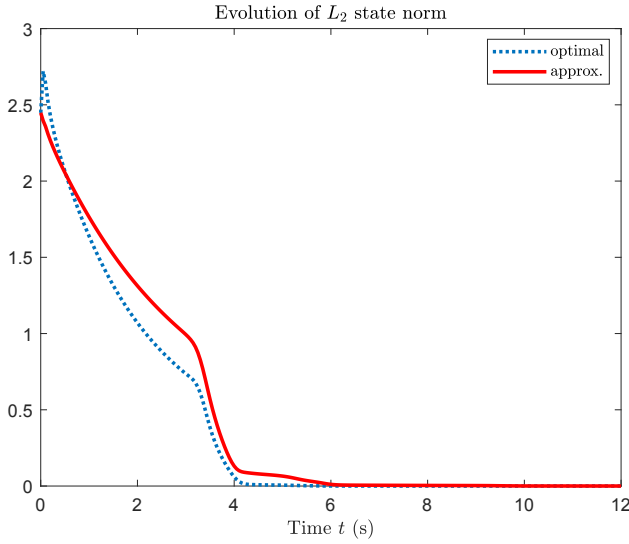


Fig. 7. Evolution of state L_2 norm.

VI. CONCLUSIONS

A computational geometry method was employed to approximate the full state control signal for a class of spatially distributed systems by the weighted sum of pointwise process measurements resulting in static output feedback control. The approximation used a modification of the Centroidal Voronoi Tessellations to partition the feedback kernel associated with a feedback operator into equiareal cells. The approximation of the full state control signal produced both the sensor locations for placing the sensors and the associated static gains. Extending to the time varying kernels, resulted in both time varying gains and time varying sensor locations. The latter rendered the sensors mobile and which required the guidance of the associated sensor network. Two different sensor guidance schemes were presented; the first one

case	$\int_0^T \ x(t)\ ^2 dt$	$\int_0^T u^2(t) dt$
full state	2.70949	52.3224
output feedback	3.79239	38.7987

TABLE I
STATE AND CONTROL ENERGY.

implemented the kernel partitioning at each time, thereby producing the requisite sensor repositioning at each time. The second one used the proposed kernel partitioning only at the initial time and subsequently used the advection partial differential equation modeling the sensor density to extract the sensor density velocity, thus propagating the mobile sensors over the spatial domain. A numerical study was provided to highlight the aspects of the proposed scheme.

Real-time implementability of the proposed kernel partitioning is a concern and will be addressed in a forthcoming work by the author.

REFERENCES

- [1] B. B. King, "Representation of feedback operators for parabolic control problems," *Proc. Amer. Math. Soc.*, vol. 128(5), pp. 1339–1346, 2000.
- [2] J. A. Burns and B. B. King, "Optimal sensor location for robust control of distributed parameter systems," in *Proceedings of the 33rd IEEE Conference on Decision and Control*, vol. 4, Dec 1994, pp. 3967–3972.
- [3] J. A. Burns and D. Rubio, "A distributed parameter control approach to sensor location for optimal feedback control of thermal processes," in *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 3, 10–12 Dec 1997, pp. 2243–2247.
- [4] A. L. Faulds and B. B. King, "Sensor location in feedback control of partial differential equation systems," in *Proc. of the IEEE Int'l Conference on Control Applications*, 25–27 Sep 2000, pp. 536–541.
- [5] I. Akhtar, J. Borggaard, J. A. Burns, H. Imtiaz, and L. Zietsman, "Using functional gains for effective sensor location in flow control: a reduced-order modelling approach," *J. of Fluid Mechanics*, vol. 781, pp. 622–656, 10 2015.
- [6] M. A. Demetriou, "Sensor selection and static output feedback of parabolic pdes via state feedback kernel partitioning using modification of voronoi tessellations," in *Proc. of the American Control Conference*, May 2017, pp. 2497–2503.
- [7] M. A. Demetriou, "Design of decentralized local controllers of spatially distributed systems by approximation of feedback kernels," in *Proc. of the American Control Conf.*, Boston, MA, July 2–6 2016.
- [8] —, "Gain adaptation and sensor guidance of diffusion pdes using on-line approximation of optimal feedback kernels," in *Proc. of the 2016 American Control Conf.*, Boston, MA, July 2–6 2016.
- [9] M. Ghanavati, A. Chakravarthy, and P. Menon, "Pde-based analysis of automotive cyber-attacks on highways," in *Proc. of the American Control Conference*, May 2017, pp. 1833–1838.
- [10] M. Ghanavati, A. Chakravarthy, and P. P. Menon, "Pde-based swarm control for contaminant tracking applications," in *Proc. of the Annual American Control Conference*, June 2018, pp. 937–942.
- [11] —, "Analysis of automotive cyber-attacks on highways using partial differential equation models," *IEEE Trans. on Control of Network Systems*, vol. 5, no. 4, pp. 1775–1786, Dec 2018.
- [12] M. Ghanavati and A. Chakravarthy, "Pde-based modeling and control for power generation management of wind farms," *IEEE Trans. on Sustainable Energy*, vol. 10, no. 4, pp. 2104–2113, Oct 2019.
- [13] R. F. Curtain and H. J. Zwart, *An Introduction to Infinite Dimensional Linear Systems Theory*. Berlin: Springer-Verlag, 1995.
- [14] H. T. Banks, R. C. Smith, and Y. Wang, *Smart Material Structures: Modeling, Estimation and Control*. New York: Wiley-Masson, 1996.
- [15] J. A. Burns and G. H. Peichl, "On robustness of controllability for finite-dimensional approximations of distributed parameter systems," in *Distributed parameter systems: modelling and simulation (Hiroshima, 1987)*. North-Holland, Amsterdam, 1989, pp. 217–222.
- [16] M. A. Celia and W. G. Gray, *Numerical methods for differential equations*. Englewood Cliffs, NJ: Prentice Hall Inc., 1992.