©Copyright 2020
Emily E. Clark

Greedy algorithms for physics-informed sparse sensor selection

Emily E. Clark

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2020

Reading Committee:

J. Nathan Kutz, Chair

Steven L. Brunton, Chair

Gerald A. Miller

Program Authorized to Offer Degree:
Physics

University of Washington

Abstract

Greedy algorithms for physics-informed sparse sensor selection

Emily E. Clark

Co-Chairs of the Supervisory Committee:
Robert Bolles and Yasuko Endo Professor, Adjunct Professor of Electrical Engineering and Physics J. Nathan Kutz
Applied Mathematics

James B. Morrison Endowed Career Development Professor in Mechanical Engineering, Adjunct Associate Professor of Applied Mathematics Steven L. Brunton
Mechanical Engineering

In this era of big data, many systems of interest to researchers are too large to fully sample. Thus, significant downsampling is necessary, but determining the best locations for optimal full-state reconstructions is an NP-hard problem. Solving for the optimal sensor selections would require the researcher to test all $\binom{n}{p}$ combinations of placing p sensors given n possible locations, which is only feasible for very small systems. Instead, researchers have developed techniques to calculate near-optimal sensor placements, usually based on convex relaxations or greedy algorithms. This text focuses on a well-known greedy algorithm, the column-pivoted QR decomposition, which is performed on basis modes from a low-rank decomposition of the system, to pick out sensor locations that are approximately maximally informative and robust to noise.

The column-pivoted QR decomposition is efficient and has proven optimality guarantees, but it does not account for several important practical considerations, including sensor cost, purpose, and type. In this work, we extend the QR decomposition to account for some of these real-world constraints. First, we modify the algorithm to account for a heterogeneous cost function on sensor location, selecting sensors that are approximately Pareto optimal in

cost and reconstruction quality. Next, we demonstrate that the cost-constrained column-pivoted QR decomposition can be applied to modal bases beyond the most common basis of singular vectors. In this way, we can select sensors and actuators for control systems, account for a system's estimated equations of motion, and even select sensors without training data. Finally, we approach the problem of multi-fidelity sensor selection, that is, determining where and how many of each type of sensor to place, given a fixed budget and access to cheap, high-noise sensors and expensive, low-noise sensors. This problem is complex and has a very large parameter space, but we develop guidelines for asymptotic cases of sensor cost and noise level. The above methods are demonstrated on examples from physics, climate science, and facial recognition, showing that it is possible to improve sensor effectiveness and decrease cost by considering real-world practicalities.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

First of all, a big thank you to my advisers J. Nathan Kutz and Steven L. Brunton, who have supported me through years of research, through getting stuck when the tasks seemed overwhelming, through my first nervous conference talks, and through the daunting process of completing this dissertation. Their unwavering optimism has been as indispensable as all the math and research skills they've taught me.

I would also like to thank my collaborator Travis Askham, who not only wrote the code for the cost-constrained QR algorithm, but also taught me how to write an academic paper.

A big shout-out to Krithika Manohar, whose work is the solid foundation for my own. And to the many students and postdocs in Nathan's and Steve's groups, thank you for keeping me in touch with all the interesting work going on in the applied math community, and for always making group meetings interesting.

I wouldn't have made it this far without my friends, thank you all for the game nights, movie nights, and for eating my obsessive baking. I would particularly like to mention my roommates, Shane Peterson and Grace Hamilton. You two have kept me sane and kept me from being a complete recluse, not to mention Grace has kept me very well fed for five years.

And finally, I wouldn't be who I am today without my wonderful family. Sean and Elise, you're two of my best friends, thank you for all the fun times and all the stupid YouTube videos. Mom, I love you. You're so strong, thank you for everything you do for this family. Dad, I wish you could have seen me graduate, but you were my role model and the best father. I love you and miss you.

I would like to acknowledge funding from Boeing, and thank them for the opportunity to work on the very interesting problem of bracket standardization. This work was also funded

by the Air Force Office of Scientific Research (FA9550-15-1-0385 and FA9550-19-1-0011), the Air Force Research Laboratory (FA8651-16-1-0003), the Air Force Office of Scientific Research through the Young Investigator Program (FA9550-18-1-0200), and a Seattle ARCS Foundation Fellowship.

DEDICATION

To my father, Kenneth Clark,

in loving memory.

Chapter 1

INTRODUCTION

Measurements have long been essential to the field of physics, from Tycho Brahe's body of astronomical data that led Kepler to his discovery that the planets follow elliptical orbits, to the experiments measuring the speed of light that led James Clerk Maxwell to prove that light is an electromagnetic wave [1]. And from the complicated relationship between a quantum system and its measurement, to detectors at the LHC that are still discovering new physics about and perhaps beyond the standard model. But many systems are too large to measure fully, including fluid flows, electrical grids, atmospheric conditions for weather prediction, and the spread of disease for epidemiology. This is particularly true as computers are becoming ever more sophisticated, allowing us to study ever larger systems, but physical sensors still cost money and have constraints on where they can be placed. Therefore *sparse sensing* is important to many diverse fields, including scientific experiments [2, 3], reduced order modeling [4, 5], control theory [6–9], climate science [10–12], robotics [13, 14], and industry [15–18]. Principled sparse sensing develops methods for the optimal selection of a limited number of sensors to obtain the best possible full-state reconstructions. A brute-force search for the selection of p sensors out of n possible locations requires checking $\binom{n}{p}$ sensor combinations, which quickly becomes intractable for even moderately large values of n and p. There are many near-optimal, more tractable solution methods, including relaxations of the combinatorially-hard full optimization problem into a convex, solvable formulation, and *greedy* solutions that select sensors one at a time, rather than searching for the global optimum. Some of these methods are briefly outlined below, but most of them fail to account for real-world considerations and restrictions on the sensors, particularly cost, noise, and

application. In this work, we use the column-pivoted QR decomposition, a fast, greedy algorithm for near-optimal sensor selection, and adapt it for practical considerations. First, we modify the algorithm to account for a heterogeneous cost function on sensor location, yielding sensor selections that approximately simultaneously minimize sensor cost and maximize reconstruction quality. Next, we apply the cost-constrained column-pivoted QR algorithm to modal decompositions beyond the usual *singular value decomposition* (SVD), expanding its capabilities to sensor selection for control systems, systems with coherent behavior in both space and time, and systems without full-state training data. Finally, we consider multi-fidelity sensor selection, and begin to develop guidelines for the selection of two types of sensors, one with low noise level and high cost, the other with high noise level and low cost. These three developments would allow researchers to place sensors more cheaply and more effectively for better real-world applicability. An outline of our approach is shown in Figure 1.1, which uses sea surface temperature snapshots to demonstrate our process of using full-state measurements to obtain a modal decomposition, combining it with a cost function on sensor location for the modified QR decomposition, and obtaining sensors with varying levels of cost and reconstruction accuracy depending on how heavily the cost function is weighted.

The rest of the text is organized as follows: this chapter concludes with a brief outline of common sparse sensing techniques. Chapter 2 introduces the sparse sensing problem setup and notation, and then the column-pivoted QR decomposition. In Chapter 3, we present an extension of the QR decomposition to select sensors that simultaneously account for reconstruction quality and sensor cost. Chapter 4 explores the optimum number of sensors and basis modes for two common modal decompositions. In Chapter 5, we go on to apply the cost-modified algorithm to other modal decompositions to best account for the dynamics of the system at hand. And we consider the selection of sensors with two different costs and noise levels in Chapter 6. Finally, conclusions and possible future work are presented in Chapter 7.

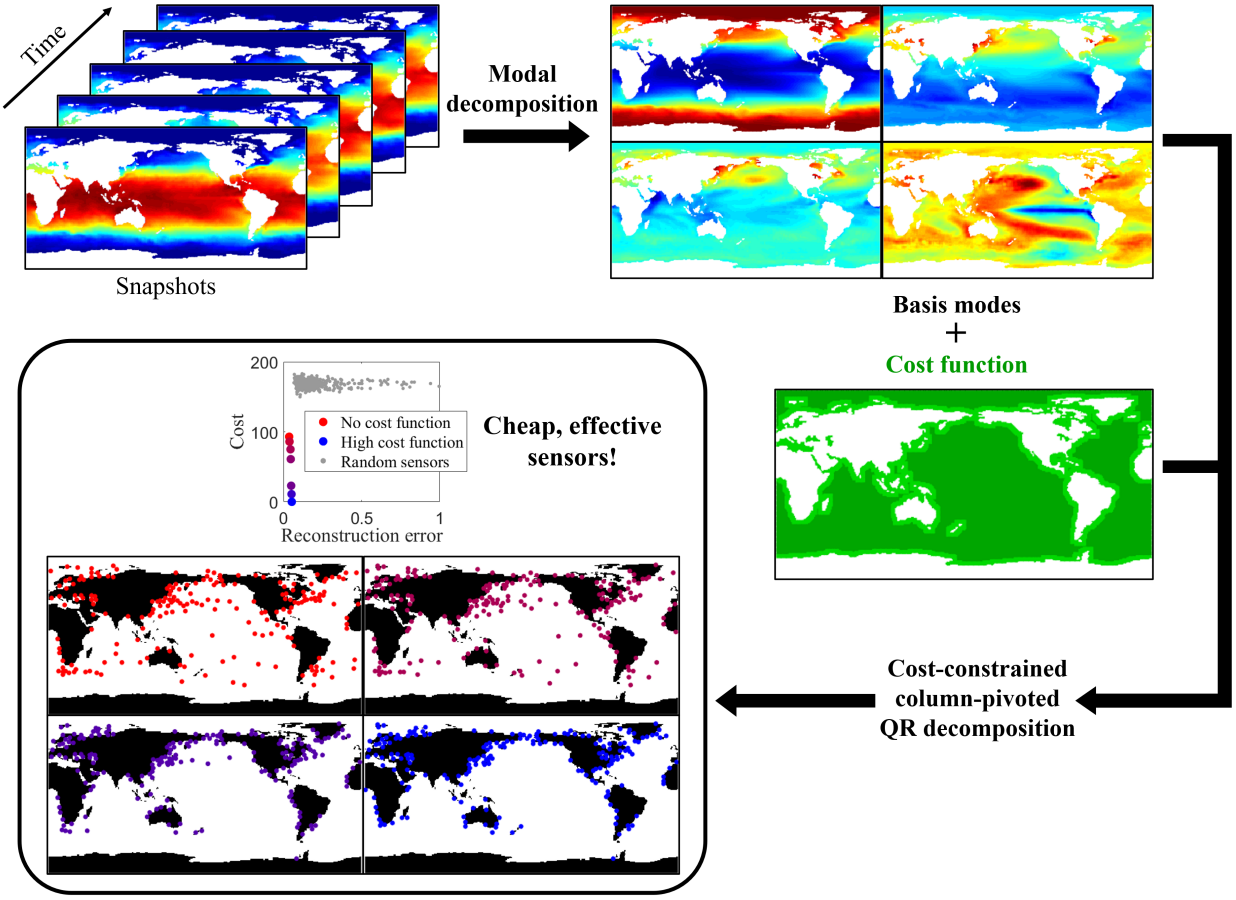

Figure 1.1: A basic illustration of our cost-constrained sensor selection procedure. Snapshots are used to obtain a modal decomposition, which, along with a cost function on sensor location, is used to obtain sensor selections. The cost-constrained column-pivoted QR algorithm yields sensors with simultaneously low cost and reconstruction error, with the balance dictated by a cost function weighting.

1.1 Background

This section reviews sparse sensing methods with similar frameworks to our own. Since the focus of this text is on full-state reconstruction, we consider the problem of finding sensor locations that minimize the reconstruction error. We assume point sensors, and obtain reconstructions by applying a linear map to the values at the sensors; this can be viewed as

an interpolation problem, where the sensors are interpolation points.

A brute-force solution of the sensor placement problem may be obtained by searching over all possible subsets of the sensors, but this approach quickly becomes intractable, as the number of subsets increases combinatorially. However, randomly placed sensors perform surprisingly well. For instance, Wright et al. observed that, given a generic basis in which samples of the signal will be sparse, it is possible to perfectly reconstruct a signal that has been downsampled or randomly projected [19]. The compressed sensing literature provides a theoretical basis for the surprising effectiveness of random, or rather *incoherent*, measurements in this setting; see, inter alia, [20–24]. Such an approach does not necessarily make use of any full-state observations of the system (though some model for the system is implied), and random sensors have been observed to be less efficient than sensors that take this data into account [25].

A common data-driven approach is to start with a *tailored basis* derived from the observed samples, typically given by the dominant singular vectors [5, 25–27]. See [4, 28] for early examples of signal reconstruction from a limited number of sensors using such a basis. A number of heuristic choices for the locations have been developed, including placing sensors at the extrema of the singular vectors [10, 29, 30].

Before continuing our discussion of sensor selection in a tailored basis, we require some notation. Assume there are m samples of data $\mathbf{x}_i \in \mathbb{R}^n$, and let these samples form the columns of a matrix $\mathbf{X}$:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \end{bmatrix}. \tag{1.1}$$

Thus the rows of $\mathbf{X}$ correspond to spatial locations. Furthermore, let $\mathbf{\Psi}$ be some basis matrix derived from $\mathbf{X}$ ($\mathbf{\Psi}$ is often taken to be the right singular vectors of $\mathbf{X}$, see Section 2.1 for a discussion of the SVD, but other bases are permitted). For an index set J, let $\mathbf{\Theta}_J$ denote the matrix formed by the rows of $\mathbf{\Psi}$ with index in J. If the size of J is fixed, it is known that the set of indices $\hat{J}$ that maximizes the product of the singular values of $\mathbf{\Theta}_J$ (or equivalently maximizes the determinant of $\mathbf{\Theta}_J$) provides optimal interpolation points for $\mathbf{\Psi}$ [31, 32].

The problem of finding such a $\hat{J}$ is nonconvex and NP-hard, but there are reasonable approximate algorithms to solve the sensor selection optimization problem. Gu and Eisenstat developed a polynomial time algorithm for computing J when the optimality criterion is relaxed slightly [31]. In [33], Li et al. considered the related problem of dual volume sampling, which would allow the placement of more sensors than there are samples of data, using a probabilistic framework. Joshi and Boyd reformulated determinant maximization as an approximate convex problem, which may be solved in polynomial time and is observed to provide nearly optimal sensors [34]. While these approaches scale polynomially in the number of sensors and the size of the data, they are not as computationally efficient as some of the existing greedy algorithms for interpolation, especially for high-dimensional data. Further, the examples on which the greedy algorithms are known to fail appear to be pathological, i.e. it is incredibly unlikely that the greedy approach will fail in practice.

The greedy sensor selection algorithm which is of greatest interest in this paper is based on the column pivoted QR decomposition. In particular, for a given number of sensors p, one simply selects J to be the first p column pivots of $\mathbf{\Psi}^*$ (see Section 2.2 for an explanation as to why this is a greedy approach for maximizing the product of the singular values of $\mathbf{\Theta}_J$). This algorithm is the basis for practical approaches to computing the interpolative decomposition [32, 35], which is commonly used to compress low-rank matrices. The algorithm is also used in the discrete empirical interpolation method (DEIM) from reduced order modeling [5, 27], in its more stable Q-DEIM formulation [36]. For high-dimensional problems with many samples of data, standard techniques from the burgeoning field of randomized algorithms for linear algebra may be used to improve the efficiency of these schemes [37, 38].

1.1.1 Submodularity

In a line of research parallel to the linear algebra based methods described above, the sensor selection problem is often formulated as submodular function optimization. A set function is submodular if it obeys a diminishing returns property: adding an element to a set produces a greater change in the function than adding the same element to a superset of that set.

Greedy methods for optimizing submodular functions are well studied; see [39] for a proof of the near-optimal performance of greedy methods for submodular functions and, inter alia, [40–46] for further developments. In particular, [41] provides a comprehensive review of submodular functions and the maximization thereof.

The topic is of interest in this work because certain performance metrics for sensor placement — including the determinant maximizing metric used here — are submodular, which means that greedy methods enjoy some optimality guarantees. For example, [47,48] leverage the submodularity of the mutual information between chosen and unchosen sensor locations to find near-optimal observation locations to model Gaussian processes. The FrameSense algorithm developed in [44] is a fast greedy method that minimizes the frame potential of the basis matrix $\mathbf{\Psi}$, a measure of the orthogonality of its rows. Under certain assumptions on the structure of the linear model, maximizing the frame potential implies a near-optimal bound on the mean square error of parameter recovery. Submodularity-based methods can be extended to include a cost function on sensor location, as in [42, 43, 49, 50].

The related problem of controlling a system based on a limited number of measurements and actuators has also been thoroughly explored, as in [6, 51–53]. Furthermore, it can be shown that certain controllability and observability metrics are submodular, and thus near-optimal sensor and actuator locations can be chosen using simple greedy algorithms [54–57]. The submodularity properties are extended to include a heterogeneous cost function in [58].

It can be verified through numerical experiment that the reconstruction error of a matrix after interpolation is not itself submodular in the interpolation points. However, the log-determinant is a good proxy for reconstruction, see [32], and is submodular. While the standard optimality guarantees either do not apply or are significantly weakened for a possibly decreasing submodular function like the log-determinant, the method is empirically effective for reconstruction, and the QR decomposition approach for computing the sensor locations is efficient and scalable. Therefore, in this text we adapt the QR decomposition approach to the cost-constrained setting, apply the adapted algorithm to dynamically relevant bases, and use it for multi-fidelity sensor selection.

1.2 Published work

The work presented here has resulted in several publications. At the time this was written, the first two papers listed have been peer reviewed and published, while the third has been peer reviewed and accepted for publication by the IEEE Sensors Journal, and is in the process of being published. The majority of the content of this thesis has been reproduced with permission from the following:

Emily Clark, Travis Askham, Steven L. Brunton, and J. Nathan Kutz. Greedy sensor placement with cost constraints. *IEEE Sensors Journal*, 19(7), 2642–2656, 2018.
©2018 IEEE.

Emily Clark, J. Nathan Kutz, and Steven L. Brunton. Sensor Selection With Cost Constraints for Dynamically Relevant Bases. *IEEE Sensors Journal*, 2020.
©2020 IEEE.

Emily Clark, Steven L. Brunton, and J. Nathan Kutz. Multi-fidelity sensor selection: Greedy algorithms to place cheap and expensive sensors with cost constraints. *arXiv preprint arXiv:2005.03650*, 2020.

Chapter 2

METHODS

This chapter introduces the sparse sensing problem and the column-pivoted QR decomposition as a greedy approximate method of solving it.

2.1 Problem formulation

Consider a high-dimensional system, such as the discretization of a PDE or a set of high-definition photographs. We wish to sparsely sample the state of the system and then reconstruct the full state from these sparse measurements. The goal is to determine the optimal sampling points that yield the best reconstructions for a given number of sensors.

Assume we have access to high-fidelity training data, either from a high-precision simulation or an experiment with full sensing. Gathering the training data is a one-time expense, and it will be used to learn the optimal sensing locations. Once m full-state snapshots $\mathbf{x}_i \in \mathbb{R}^n$ have been collected, they are gathered into a data matrix

$$\mathbf{X}^{tr} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \end{bmatrix}. \tag{2.1}$$

With this training data, we construct a modal basis to represent the system, such that any snapshot $\mathbf{x}_i$ can be constructed from the decomposition

$$\mathbf{x}_i = \mathbf{\Psi}\mathbf{a}_i, \tag{2.2}$$

where the columns of $\mathbf{\Psi} \in \mathbb{R}^{n\times r}$ are the basis modes, and $\mathbf{a}_i \in \mathbb{R}^r$ are the coefficients of $\mathbf{x}_i$ in the $\mathbf{\Psi}$ basis. Usually the number of basis modes $r < \min(m, n)$, so that $\mathbf{\Psi}$ is a *reduced basis*, and often $\mathbf{\Psi}$ is taken to be the first r spatial modes of the singular value decomposition (SVD)

of $\mathbf{X}^{tr}$. The SVD is analogous to an eigenvalue decomposition for non-square matrices, and is given by

$$\mathbf{X}^{tr} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*, \tag{2.3}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal, the columns of $\mathbf{U}$ are the coherent spatial modes of $\mathbf{X}^{tr}$, the columns of $\mathbf{V}$ are the temporal modes, and $\mathbf{\Sigma}$ is diagonal with entries $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$. The σ_i are the singular values, and they denote the amount of energy contained in the i^{th} mode.

For reduced order modeling, it is very common to take

$$\mathbf{\Psi} = \mathbf{U}_{:,1:r}, \tag{2.4}$$

where $\mathbf{U}_{:,1:r}$ is the first r columns of $\mathbf{U}$, which provides the optimal rank-r approximation of a data set. That is, by the Eckart-Young theorem [59],

$$\mathbf{U}_{:,1:r} = \underset{\mathbf{\Psi},\ \text{rank}\mathbf{\Psi}=r}{\arg\min} ||\mathbf{X}^{tr} - \mathbf{\Psi}\mathbf{\Psi}^*\mathbf{X}^{tr}||_F, \tag{2.5}$$

where $||\cdot||_F$ is the Frobenius norm.

This is the default choice for most sparse sensing applications, but there is flexibility to choose the basis that is right for the system and application at hand, see e.g. [60] and Chapter 5.

Now let $\mathbf{X}^{te} \in \mathbb{R}^{n\times\ell}$ be a test set of ℓ new snapshots to be downsampled. Given $p \ll n$ sensors, we must select p locations to measure, indexed by the set J, where $|J| = p$ and for all indices $j \in J$, $1 \leq j \leq n$. We downsample using the selection matrix $\mathbf{C}_J \in \mathbb{R}^{p\times n}$, where the rows of $\mathbf{C}_J$ are the unit vectors $\mathbf{e}_J^T$. Thus the measurements $\mathbf{Y}(J) \in \mathbb{R}^{p\times\ell}$ are given by

$$\mathbf{Y}(J) = \mathbf{C}_J\mathbf{X}^{te} \tag{2.6a}$$

$$= \mathbf{C}_J\mathbf{\Psi}\mathbf{a}^{te} \tag{2.6b}$$

$$= \mathbf{\Theta}_J \mathbf{a}^{te}, \tag{2.6c}$$

where $\mathbf{a}^{te} \in \mathbb{R}^{r \times \ell}$ are the coefficients of the test set in the $\mathbf{\Psi}$ basis, and $\mathbf{\Theta}_J = \mathbf{C}_J \mathbf{\Psi}$ is the measurement matrix, consisting of the rows of $\mathbf{\Psi}$ that are in J.

To perform full-state reconstruction given sparse measurements, we take the minimum norm least squares solution,

$$\hat{\mathbf{a}}^{te}(J) = \mathbf{\Theta}_J^{\dagger} \mathbf{Y}(J) \tag{2.7}$$

$$\hat{\mathbf{X}}^{te}(J) = \mathbf{\Psi} \hat{\mathbf{a}}^{te}(J), \tag{2.8}$$

where $\mathbf{\Theta}_J^{\dagger}$ is the pseudoinverse of $\mathbf{\Theta}_J$. The fractional reconstruction error for the system given the measurements J is

$$E(J) = \frac{||\mathbf{X}^{te} - \hat{\mathbf{X}}^{te}(J)||_F}{||\mathbf{X}^{te}||_F} \tag{2.9a}$$

$$= \frac{||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Theta}_J^{\dagger}\mathbf{C}_J\mathbf{X}^{te}||_F}{||\mathbf{X}^{te}||_F}. \tag{2.9b}$$

With this performance metric, the sparse sensor selection problem is to find the optimal index set $\hat{J}$ such that

$$\hat{J} = \arg\min_J E(J). \tag{2.10}$$

This is a combinatorially hard problem, and we employ a greedy solution, as described in the next section.

2.2 Column-pivoted QR decomposition

Our greedy sensor selection method is the column-pivoted QR decomposition, as described in [36]. The QR decomposition was introduced for the solution of least squares problems in 1965 by Businger and Golub [61], and in 2016, Drmač and Gugercin [36] showed that the greedy determinant maximization of the column-pivoted QR algorithm is highly effective for

sensor placement. For a given matrix $\mathbf{V} \in \mathbb{R}^{r\times n}$, $r \leq n$, the QR decomposition with column pivoting is given by:

$$\mathbf{VP} = \mathbf{QR}, \tag{2.11}$$

where $\mathbf{Q} \in \mathbb{R}^{r\times r}$ is unitary, $\mathbf{R} \in \mathbb{R}^{r\times n}$ is upper triangular, and $\mathbf{P} \in \mathbb{R}^{n\times n}$ is a column permutation matrix.

The procedure for the column-pivoted QR decomposition with Householder reflections is as follows: First initialize the permutation matrix to $\mathbf{P}_0 = \mathbb{I}_n$, then select the column $\mathbf{v}_j$ of $\mathbf{V}$ with the largest norm and swap it with the first column of $\mathbf{V}$ via the permutation matrix:

$$\mathbf{P}_1 = \begin{pmatrix} \mathbf{e}_j & \mathbf{e}_2 & \cdots & \mathbf{e}_{j-1} & \mathbf{e}_1 & \mathbf{e}_{j+1} & \cdots & \mathbf{e}_n \end{pmatrix}. \tag{2.12}$$

Next apply a Householder transformation [62] $\mathbf{Q}_1$, a unitary transformation that reflects $\mathbf{v}_j$ onto the first unit vector $\mathbf{e}_1$. If $\mathbf{u} = \mathbf{v}_j - ||\mathbf{v}_j||\mathbf{e}_1$ is the hyperplane that bisects $\mathbf{v}_j$ and $\mathbf{e}_1$, and $\mathbf{w} = \frac{\mathbf{u}}{||\mathbf{u}||}$, then the first Householder reflection $\mathbf{Q}_1$ is given by

$$\mathbf{Q}_1 = \mathbb{I}_n - 2\mathbf{w}\mathbf{w}^*, \tag{2.13}$$

and

$$\mathbf{Q}_1\mathbf{v}_j = \begin{pmatrix} ||\mathbf{v}_j|| \\ 0 \\ \vdots \end{pmatrix}. \tag{2.14}$$

Apply the Householder reflection to $\mathbf{VP}_1$ to obtain

$$\mathbf{Q}_1\mathbf{VP}_1 = \begin{pmatrix} ||\mathbf{v}_j|| & * & * & \cdots \\ 0 & & & \\ 0 & & \mathbf{V}' & \\ \vdots & & & \end{pmatrix}. \tag{2.15}$$

Algorithm 1 QR pivoting.

Input: matrix $\mathbf{V} \in \mathbb{R}^{r \times n}$, number of sensors $p \leq r$
Output: Partial permutation matrix $\mathbf{P}$

1: $\mathbf{P} \leftarrow \mathbb{I}_n$
2: **for** $k = 1, \ldots, p$ **do**
3: $\quad j_k \leftarrow \arg\max_{j \geq k} ||\mathbf{V}_{k:r,j}||$
4: $\quad$ `Swap`$(\mathbf{P}_{:,k}, \mathbf{P}_{:,j_k})$.
5: $\quad$ Calculate Householder reflection $\mathbf{Q}_k$ such that $\mathbf{Q}_k \begin{pmatrix} V_{k,j_k} & V_{k+1,j_k} & \cdots & V_{r,j_k} \end{pmatrix}^T = \begin{pmatrix} ||\mathbf{V}_{k:r,j_k}|| & 0 & \cdots & 0 \end{pmatrix}^T$.
6: $\quad \mathbf{V} \leftarrow \mathrm{diag}(\mathbb{I}_{k-1}, \mathbf{Q}_k)\mathbf{V}\mathbf{P}$
return $\mathbf{P}_{:,1:p}$

Repeat the process on the submatrix $\mathbf{V}'$, i.e. identify the column $\mathbf{v}'_{j'}$ with the largest norm, swap it with the first column of $\mathbf{V}'$, and transform it so that its subdiagonal elements are zero:

$$\mathbf{Q}_2\mathbf{Q}_1\mathbf{V}\mathbf{P}_2 = \begin{pmatrix} ||\mathbf{v}_j|| & \star & \star & \cdots \\ 0 & ||\mathbf{v}'_{j'}|| & \star & \cdots \\ 0 & 0 & & \\ \vdots & \vdots & & \mathbf{V}'' \end{pmatrix}. \tag{2.16}$$

Continue to iterate until the right-hand side is upper triangular and the decomposition is achieved:

$$\mathbf{Q}_r \cdots \mathbf{Q}_1\mathbf{V}\mathbf{P}_r = \mathbf{R}. \tag{2.17}$$

Thus $\mathbf{P}$ from Equation 2.11 is $\mathbf{P}_r$, and $\mathbf{Q} = \mathbf{Q}_1^* \cdots \mathbf{Q}_r^*$.

Note that the k^{th} Householder transformation is acting on a vector of length $r - k + 1$, and so $\mathbf{Q}_k$ is augmented by the appropriate identity matrix. Also see Algorithm 1.

This algorithm imposes a diagonal dominance structure in $\mathbf{R}$, i.e.

$$|r_{ii}|^2 \geq \sum_{j=1}^{k} |r_{jk}|^2, \qquad 1 \leq i \leq k \leq r, \tag{2.18}$$

and therefore maximizes the determinant of the $r \times r$ left-hand submatrix of $\mathbf{R}$. In fact,

because the Householder transformations are unitary, at every iteration this procedure maximizes the determinant of the upper left submatrix that has been made upper triangular.

To perform sensor selection, we apply the column-pivoted QR algorithm to $\mathbf{\Psi}^*$ and take $\mathbf{C}_J^T = \mathbf{P}_{:,1:p}$, where $\mathbf{P}_{:,1:p}$ are the first p columns of $\mathbf{P}$. In other words, the sensor locations correspond to the rows of $\mathbf{\Psi}$ with the iteratively largest norms, such that the determinant of $\mathbf{\Theta}_J$, those selected rows, is maximized. Or, to generalize to the case of $p \neq r$ (i.e. $\mathbf{\Theta}_J$ is nonsquare), equivalently the determinant of $\mathbf{\Theta}_J^* \mathbf{\Theta}_J$ is maximized.

Thus, QR with column pivoting does not solve Equation 2.10, but rather it greedily solves

$$\arg\max_J \left|\det \mathbf{\Theta}_J^* \mathbf{\Theta}_J\right|, \tag{2.19}$$

an approach that is equivalent to D-optimal experimental design [63], which seeks to maximize $\det \mathbf{\Theta}_J^* \mathbf{\Theta}_J$. Other optimal experimental design approaches include A-optimal design, which minimizes the trace of $(\mathbf{\Theta}_J^* \mathbf{\Theta}_J)^{-1}$ and therefore the variance of $\hat{\mathbf{a}}^{te}$, and E-optimal design, which maximizes the smallest eigenvalue of $\mathbf{\Theta}_J^* \mathbf{\Theta}_J$ for stable reconstructions.

Determinant maximization maximizes the volume of the measurement matrix, ensuring that it selects informative locations. Moreover, it causes $\mathbf{\Theta}_J$ to be stable under inversion by minimizing its condition number

$$\kappa = \frac{\sigma_{max}}{\sigma_{min}}, \tag{2.20}$$

where σ are the singular values of $\mathbf{\Theta}_J$. This is achieved because for a square $r \times r$ matrix $\mathbf{\Theta}$,

$$\left|\det \mathbf{\Theta}\right| = \prod_{i=1}^{r} \sigma_i, \tag{2.21}$$

and the condition number minimization also generalizes to the case of a non-square $\mathbf{\Theta}_J$. A large condition number leads to amplification of any noise in the signal when the full-state reconstruction is performed, so although we have not directly solved the optimization problem in terms of reconstruction error, we can be assured that the column-pivoted QR decomposition selects desirable sensor locations.

Although the greedy QR decomposition algorithm does not necessarily converge to the optimal solution of either Equation 2.10 or 2.19, it is easy and fast to implement, being included on any standard software package and requiring $\mathcal{O}(nrp)$ operations (note that typically the vectors corresponding to the Householder reflector are stored rather than the matrix $\mathbf{Q}$ itself in an efficient implementation). We find empirically that it performs well in practice when compared to random sensor selections. We also highlight the work of Drmač and Gugercin [36] who showed the algorithm to be highly effective for sensor placement.

Notice that our problem setup and Algorithm 1 do not require $\mathbf{\Psi}$ to be orthogonal, which will allow us to apply QR pivoting to any appropriate basis for sensor selection, as demonstrated in the following chapters.

Chapter 3

GREEDY SENSOR SELECTION WITH COST CONSTRAINTS

The problem of optimally placing sensors under a cost constraint arises naturally in the design of industrial and commercial products, as well as in scientific experiments. In this chapter, we extend the greedy column-pivoted QR algorithm to account for a cost function on sensor location, and demonstrate the effectiveness of the new algorithm on data sets related to facial recognition, climate science, and fluid mechanics. This algorithm is scalable and often identifies sparse sensors with near-optimal reconstruction performance, while dramatically reducing the overall cost of the sensors. We find that the cost-error landscape varies by application, with intuitive connections to the underlying physics. Additionally, we include experiments for various pre-processing techniques and find that a popular technique based on the singular value decomposition is often suboptimal.

3.1 Introduction

Determining the optimal placement of sensors under a cost constraint is relevant to many fields of scientific research and industry. Indeed, such considerations are critical in evaluating global monitoring systems and characterizing spatio-temporal dynamics (e.g. the brain, ocean and atmospheric dynamics, power grid networks, fluid flows, etc). For these applications, it is typical that only a limited number of measurements can be made of the system due to either prohibitive expense (i.e. either sensors are expensive, or they are expensive to place, or both) or the inability to place a sensor in a desired location (inaccessibility).

There are various high-level objectives for sensor selection, most of which are well studied. Common objectives include classification [64,65], reconstruction [4,10,25,28–30,34], reduced-order modeling [4,5,27,36], and control [56,58]. In contrast, the case in which different sensor

locations may have different associated costs has received relatively little attention, with the notable exception of the submodularity literature [41–46, 49, 50, 58]. As reconstruction error is not itself submodular (see Section 1.1.1 for more on submodularity), the goal of this chapter is to develop an algorithm that directly targets the reconstruction objective, while incorporating heterogeneous constraints on the measurement locations.

To this end, we modify the column-pivoted QR algorithm to develop a heuristic, greedy sampling strategy whereby the sensor selection optimization is formulated as a cost-constrained problem in a relaxed form.[1] [25, 36]. We further introduce a parameter representing the balance between the quality of the reconstruction and the cost, and thus can evaluate a cost-error curve.

In the techniques described in Section 1.1, an optimal map and set of sensor locations are found for a fixed number of sensors. This is equivalent to the cost-constrained sensor selection problem when each sensor has the same cost. In the case that some sensor locations should be entirely excluded, corresponding to an infinite cost, i.e. an inaccessible measurement location, and the remaining locations are of uniform cost, again the algorithms above may be used by simply narrowing the search to the allowed sensor locations (note that such a restriction has implications for the stability of the interpolation map).

We show that it is simple to modify the pivoted QR decomposition based scheme to incorporate a cost constraint for problems in between these extremes, i.e. for problems in which some sensor locations cost more than others but may be more informative. The method is obtained by writing the cost-constrained problem in a relaxed form, which introduces a parameter representing the balance between the quality of the reconstruction and the cost, and then varying that parameter to trace out a cost-error curve. For each value of the parameter, we use a greedy algorithm to add sensor locations one by one.

We test the performance of our method on data sets from facial recognition, climate sci-

[1] MATLAB code for our algorithm can be found at `https://github.com/askhamwhat/sensors-cost-paper`, along with a simple example of its use and codes to make some of the figures from Section 3.5.

ence, and fluid mechanics using a standard training-set/testing-set apparatus. In many cases, the proposed algorithm displays a significant advantage over methods based on randomly selected sensors. We also compare with known performance bound approximations where applicable and find that our algorithm is often near the approximately optimal solution.

Furthermore, we include a brief discussion of the effect of data pre-processing (i.e. the choice of $\mathbf{\Psi}$) on the quality of the sensor locations; in particular, we compare the performance when applied to the raw data, the first several singular vectors of the data, and randomized projections of the data. We find that with the unmodified column-pivoted QR decomposition, using the raw data (taking $\mathbf{\Psi} = \mathbf{X}$) yields the lowest reconstruction error at a given number of sensors, while randomized projections of the data lead to slightly higher errors at a reduced computational cost. Taking the first several singular vectors of $\mathbf{X}$ gives the highest reconstruction errors of the three pre-processing methods tested. The results are the same when a cost function is incorporated using our modified QR decomposition, but additionally we note that all three pre-processing techniques yield comparable costs.

The remainder of this chapter is organized as follows: We develop an algorithm for cost-constrained sensor selection in Section 3.2 which is a simple extension of existing methods. Section 3.3, describes the three data sets used as examples. We discuss pre-processing in Section 3.4, then apply our method to the three data sets and discuss the performance in Section 3.5. Finally, some concluding remarks are given in Section 3.6.

3.2 Algorithm for sensor selection under cost constraints

We now define the sensor selection problem with cost constraints. Let $\boldsymbol{\eta} \in \mathbb{R}^n_+$ be the costs associated with each sample location and let B denote the desired upper bound on the budget. Then the cost-constrained problem can be written as

$$\hat{J} = \arg\min_J E(J) \text{ s.t. } \sum_{j \in J} \eta_j \leq B. \tag{3.1}$$

Our algorithm will actually focus on the following relaxation of (3.1). Note that, for a given B, there exists a λ such that the problem

$$\hat{J} = \arg\min_{J} E(J) + \lambda \sum_{j \in J} \eta_j \tag{3.2}$$

and (3.1) have the same solution. Because we are often interested in the cost-error landscape, we seek the solution of (3.2) for a number of values of λ, so that a cost-error curve is traced out. We note that the main algorithm presented in this chapter does not actually solve (3.2). Instead, it provides a greedy approximate solution that uses a heuristic strategy to bias the sensors in favor of stability.

We can now derive a greedy algorithm for the relaxed version of the cost-constrained sensor selection problem (3.2). The algorithm is based on the column-pivoted QR algorithm described in Section 2.2, where the pivot column is now chosen to balance the decrease in the error $E(J)$ with the increase in the total cost $\sum_{j \in J} \eta_j$.

Let $(\mathbf{\Psi}^*)^{(k)}$ denote the lower right submatrix of a partial QR factorization of $\mathbf{\Psi}^*$ that has not yet been made upper triangular at the k^{th} iteration, and let $\boldsymbol{\eta}^{(k)}$ be the last $n-k$ elements of $\mathbf{P}_k \boldsymbol{\eta}$, where $\mathbf{P}_k$ is the permutation matrix at the k^{th} step. That is, $\boldsymbol{\eta}^{(k)}$ is the subset of $\boldsymbol{\eta}$ with indices that have not yet been selected as column pivots.

To incorporate the effect of the cost of a sensor, we update the factorization by finding the j^{k+1} that satisfies

$$j^{k+1} = \arg\max_{j} || \, (\mathbf{\Psi}^*)_j^{(k)} \, || - \gamma \eta_j^{(k)}. \tag{3.3}$$

where $\gamma \geq 0$ is some fixed parameter in the optimization. This process is summarized in pseudocode in Algorithm 2.

The scalar weighting γ simultaneously converts the units of $\boldsymbol{\eta}$ to those of $||\mathbf{\Psi}^*||$ and determines the relative importance of cost and reconstruction accuracy. When $\gamma = 0$, the pivot chosen by the formula (3.3) is the same as the pivot in column-pivoted QR; a very large value of γ approaches the minimum-cost sensor array. What constitutes a small or large value of γ depends on both the system and the cost function, and we select a set of

Algorithm 2 QR pivoting with cost function.

Input: matrix $\mathbf{V} \in \mathbb{R}^{r \times n}$, cost function $\boldsymbol{\eta} \in \mathbb{R}^n$, weighting γ, number of sensors $p \leq r$
Output: Partial permutation matrix $\mathbf{P}$

1: $\mathbf{P} \leftarrow \mathbb{I}_n$
2: **for** $k = 1, \ldots, p$ **do**
3: $\quad j_k \leftarrow \arg\max_{j \geq k} ||\mathbf{V}_{k:r,j}|| - \gamma \eta_j$
4: $\quad \texttt{Swap}(\mathbf{P}_{:,k}, \mathbf{P}_{:,j_k})$.
5: $\quad$ Calculate Householder reflection $\mathbf{Q}_k$ such that $\mathbf{Q}_k \begin{pmatrix} V_{k,j_k} & V_{k+1,j_k} & \cdots & V_{r,j_k} \end{pmatrix}^T = \begin{pmatrix} ||\mathbf{V}_{k:r,j_k}|| & 0 & \cdots & 0 \end{pmatrix}^T$.
6: $\quad \mathbf{V} \leftarrow \text{diag}(\mathbb{I}_{k-1}, \mathbf{Q}_k)\mathbf{V}\mathbf{P}$
7: $\quad \boldsymbol{\eta} \leftarrow \mathbf{P}\boldsymbol{\eta}$
return $\mathbf{P}_{:,1:p}$

γ values empirically so as to trace out a cost versus error curve. It will be shown in the following sections and chapters that this method often approximately follows the Pareto front of simultaneously minimum reconstruction error (or maximum performance metric, where applicable) and minimum cost, when compared to random sensor arrays.

Note that in the case of completely inaccessible locations (equivalent to infinite cost), the unmodified QR decomposition can be performed on the subset of allowed sensor locations.

Remark 1 *The pivot as chosen in* (3.3) *does not necessarily correspond to the natural greedy choice, i.e. the pivot that minimizes* $E(J^k) + \gamma \sum_{j \in J^k} \eta_j$ *with* J^{k-1} *fixed. Such a pivot could be computed, though at greater cost than the present algorithm. Further, there is another reason to avoid such a strategy: it completely ignores the stability of the resulting map. By instead pivoting based on column size, we bias the algorithm toward choosing stable pivots while still incorporating some sense of the reduction in error.*

3.3 Description of data sets

In this section, we provide details about the three example data sets considered below, as well as the cost functions and parameters used. An example snapshot from each system is given in Figure 3.1.

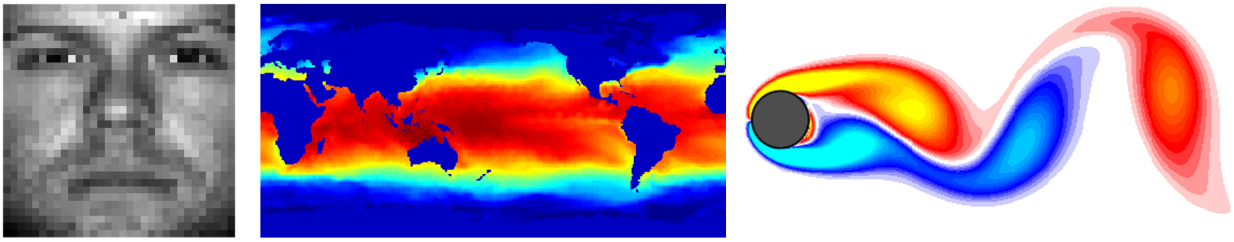

Figure 3.1: An example from each of the three data sets considered in this work. Pictured from left to right are one of the Yale B faces, a snapshot of global sea surface temperature, and a snapshot of vortex shedding of a fluid flowing around a stationary cylinder.

Two types of training set will be considered: interpolative and extrapolative. By interpolative, we mean that we have sampled a subset of the data that draws from all regions of the parameter space. By extrapolative, we mean that we have purposefully missed data from a portion of the parameter space. The data that is left out forms the testing set. Finding good sensor locations for the extrapolative training sets is a harder problem; the sensor locations must reasonably generalize to samples of data which may be unlike anything in the training set.

The first example is the Extended Yale Face Database B, referred to here as the eigenface data set [66–69]. It comprises about 64 images each of 38 individuals under various lighting conditions. The images are downsized to 32×32 pixels. Unless otherwise stated, the tests are conducted on an interpolative training set, by randomly selecting 80% of the images. A 20-fold cross validation is performed. When an extrapolative training set is used, the algorithm is trained on the first 80% of the images, meaning that some individuals appear only in the test set. Three cost functions are tested for the eigenfaces: (i) a Gaussian function, such that it is most expensive to place sensors in the center of the face, (ii) a step function uniformly penalizing sensors in the middle ninth of the photographs, and (iii) a step function penalizing the left third of the images. All three cost functions are plotted in the left column of Figure 3.4. The range of values used for the parameter γ is dependent on the cost function

considered. With the Gaussian cost function, γ ranges from 0 to 1.9×10^5, while the two step functions both take γ between 0 and 6×10^4.

The next data set we consider is the NOAA_OISST_V2 mean sea surface temperature set [70–72], comprising weekly global sea surface temperature measurements between the years of 1990 and 2016. There are a total of 1400 snapshots on a 360×180 spatial grid. The algorithm is trained on 1100 snapshots (randomly selected for the interpolative training set, or the first 1100 in sequence for the extrapolative set) and tested on the remaining 300. Ten cross validations are performed for the interpolative trials. The cost function used is a step function that penalizes placing sensors too far from shore, being zero for locations one and two pixels off land, and equal to one everywhere else. Values of γ range from 0 to 225.

Our final example is the vortex shedding of a fluid flowing around a stationary cylinder. The flow data is generated using the immersed boundary projection method [73, 74] to numerically simulate the Navier-Stokes equations with Reynolds number 100. There are 151 snapshots in time, each on a 199×449 spatial grid. The training sets consist of 120 snapshots. As before, these snapshots are chosen randomly for the interpolative training set, while the first 120 are taken in sequence to form the extrapolative training set. In the interpolative case, 30 cross validations are performed. We choose a cost function that is uniformly equal to one in the lower half of the domain and zero in the upper half, to exploit the symmetry of the system. Values of γ are between 0 and 15.

These data sets were chosen because they represent classic examples from several branches of the sensor selection literature. The face data has been used to demonstrate classification methods by discriminating between men and women, or between individuals, as in [19, 65, 75]. Sensor selection for ocean data reconstruction has a long history, including the papers [10, 29, 30], in which sensors were placed at the extrema of the SVD modes. The flow behind a stationary cylinder is a prototypical example from the reduced order modeling community, which often uses the empirical interpolation method [27] or the discrete empirical interpolation method (DEIM) [5, 36] to create a low-rank model of a system for future state prediction. See [25] and the citations therein for a more in-depth exploration of the data sets

and how they have been analyzed historically. Some of the existing methods, such as gappy POD and DEIM, could be adapted to include a heterogeneous sensor cost function; however, we have chosen to adapt the pivoted QR algorithm because it is robust, computationally efficient, and when unmodified it provides near-optimal results [25, 32, 36]. Many of the existing methods for these problems perform some pre-processing, often by applying the algorithm to singular vectors of the data rather than the data directly, a concept which we will explore in the next section.

Remark 2 *The weighting γ is an arbitrary scalar, and its values were chosen by trial and error to ensure that a cost-error curve was traced out in sufficient detail for each data set and cost function.*

3.4 Data, singular vectors, and random projections

Before proceeding to the cost-constrained selection examples, we will first briefly discuss the question of data pre-processing for sensor selection. In the notation of Section 2.1, pre-processing refers to the process of creating the matrix $\mathbf{\Psi}$ from the training data $\mathbf{X}^{tr}$ (we then apply the QR-based algorithm to $\mathbf{\Psi}^*$). When selecting p sensors, a common choice for the matrix $\mathbf{\Psi}$ is to set it as the first p right singular vectors of $\mathbf{X}^{tr}$ [5, 25, 36]. Inspired by the randomized linear algebra community [37, 38], we also consider setting $\mathbf{\Psi} = \mathbf{X}^{tr}\mathbf{G}$, where the entries of $\mathbf{G} \in \mathbb{R}^{m\times 2r}$ are drawn from a standard normal distribtuion, i.e. the columns of $\mathbf{\Psi}$ are random linear combinations of the columns of $\mathbf{X}^{tr}$. We also consider the performance when setting $\mathbf{\Psi} = \mathbf{X}^{tr}$, i.e. the performance without pre-processing. The number of singular vectors or random linear combinations used is open-ended; we found that, as described above, p singular vectors and $2p$ random linear combinations gave reasonably optimal performances for each pre-processing technique, respectively.

Our data sets are the Extended Yale Face Database B, the Optimally Interpolated Sea Surface Temperature data set from NOAA, and simulation data for fluid flow behind a cylinder. For the face and sea surface temperature data, we consider both interpolative and

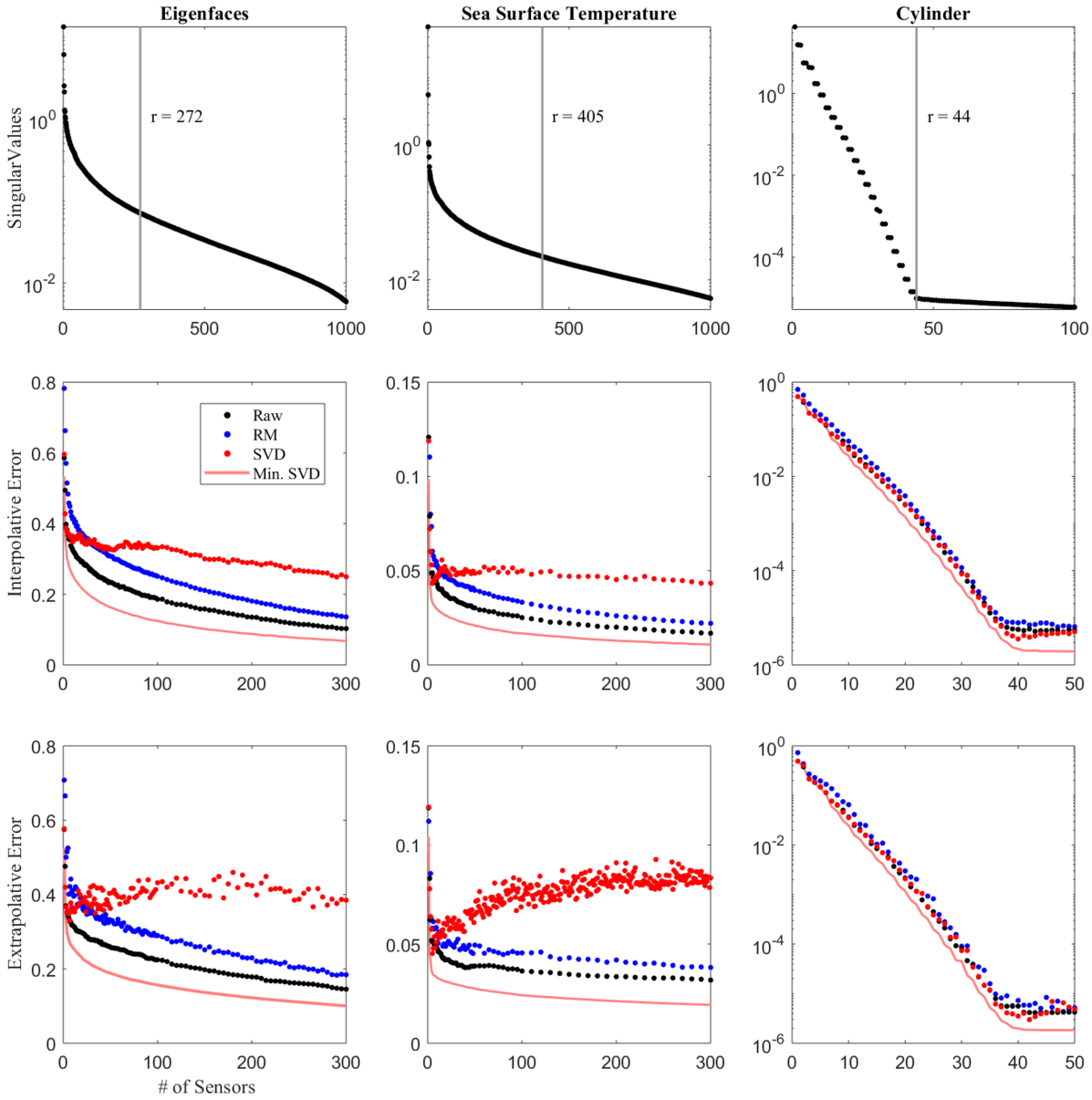

Figure 3.2: Reconstruction error versus the number of sensors for the three data sets described in Section 3.3, using p SVD modes, $2p$ randomized linear combinations (abbreviated RM in the legend, for randomized modes), and the raw data without pre-processing. The top row shows a log plot of the normalized singular values, with the vertical gray line indicating the Gavish-Donoho cutoff r [76]. The remaining plots show the average reconstruction error given sensors placed by performing the unmodified column-pivoted QR decomposition on $\mathbf{\Psi}$ formed using the three pre-processing methods discussed in the text. The first column provides eigenface results, the second gives sea surface temperature reconstruction errors, and the third shows errors for the flow behind a cylinder on a log scale. The middle row of the figure shows interpolative error, where the training set consists of a randomly-chosen subset of the data, while the bottom row gives the extrapolative data, which trains on the first 80% of the parameter space. All plots also show a rough estimate of the minimum error at a given number of sensors (the solid red line), obtained by projecting the full image onto the SVD modes.

extrapolative training sets. In this section, we do not make such a distinction for the fluid simulation data, as it is much lower rank and periodic in time.

The top row of Figure 3.2 shows the spectrum of normalized singular values for all three data sets. These plots include a gray line at the Gavish-Donoho [76] optimal hard-threshold cutoff, which is an estimate of the rank beyond which the SVD modes represent additive noise. It is apparent that the fluid flow data set is fundamentally different from the other two, having a sharp elbow at the cutoff, as opposed to a slow decay.

The remainder of Figure 3.2 plots the relative interpolation error (2.9b) computed for the test set as a function of the number of sensors using each of the three methods for pre-processing (SVD modes, randomized modes, and the raw data) described above. The sea surface temperature and eigenface data sets both have an interesting feature not present in the fluid flow data: the error from the SVD basis has a local minimum at a very small number of modes — five for the temperature data and approximately seventy for the eigenfaces. While the error for the interpolative training set begins to slowly decrease again as the number of sensors is increased, the performance of the extrapolative data gets worse as more sensors are added, up until at least 150 sensors. These results are corroborated by the work of Peherstorfer et al [77], who determine that the increase in error is due to noise amplification as sensors are added, and that the error increases $\sim \sqrt{p}$. They prove that the reconstructions can be stabilized through oversampling, taking $p > r$, but their work was completed after that of [78], which forms the basis for this chapter. Therefore, for the remainder of this chapter, we will employ a randomized basis with $r > p$. In Chapter 4, we will undertake a more thorough comparison of an SVD basis with oversampling and a randomized basis with undersampling, and find that the two methods yield nearly identical results.

The reconstruction errors for sensors based on random linear combinations or the raw data do not exhibit noise amplification (except for a weak effect with extrapolative sea surface temperature data), nor do any of the cylinder trials. Indeed, the error for the random linear combinations and the raw data behaves as expected, decaying at a rate that follows the error obtained from projecting the test set onto the first p singular modes of the training set (this

Color	Method	Description
●	Raw, QR	Performing the QR-based algorithm directly on the raw data.
●	Raw, RS	Randomly-placed sensors, using the raw data as a basis.
●	RM, QR, 2p	The QR-based algorithm on the randomized modes $\mathbf{\Psi}^{tr} = \mathbf{G}\mathbf{X}^{tr}$, $\mathbf{G} \in \mathbb{R}^{2p \times m}$ a matrix with randomized entries.
●	RM, RS, 2p	Using randomly-selected sensor locations, and $2p$ randomized modes as a basis.
●	SVD, QR, 1p	The QR-based algorithm on the first p SVD modes.

Table 3.1: A brief description of the sensor selection methods used to create Figure 3.3, and the colors in which they are plotted.

rough indicator of optimal behavior is plotted as a solid red line in the figure).

We make a few conclusions based on these pre-processing results. If the goal of pre-processing is to improve the quality of the sensors, then it appears that using no pre-processing is the preferred method, except when placing a very small number of sensors, where the SVD mode method displays an advantage. If the goal of the pre-processing is to reduce the size of the problem and speed up the optimization procedure, then it appears that using randomized linear combinations of the modes is preferable to using a limited number of SVD modes. (Note that these SVD modes would have to be computed with an accelerated procedure in order for using them to represent a speed-up for the QR-based sensor selection algorithm, again, with the caveat that SVD modes behave better for a small number of sensors.)

3.5 Applications

When factoring in the effect of cost, the behavior is similar to the cost-free case analyzed in the previous section, with respect to pre-processing. Figure 3.3 provides an overview of the performance for the various pre-processing methods, now with a non-zero cost associated with each location. In the plots, we generate several cost, error pairs for each pre-processing technique by varying the cost function weighting γ in the QR-based algorithm. We also plot

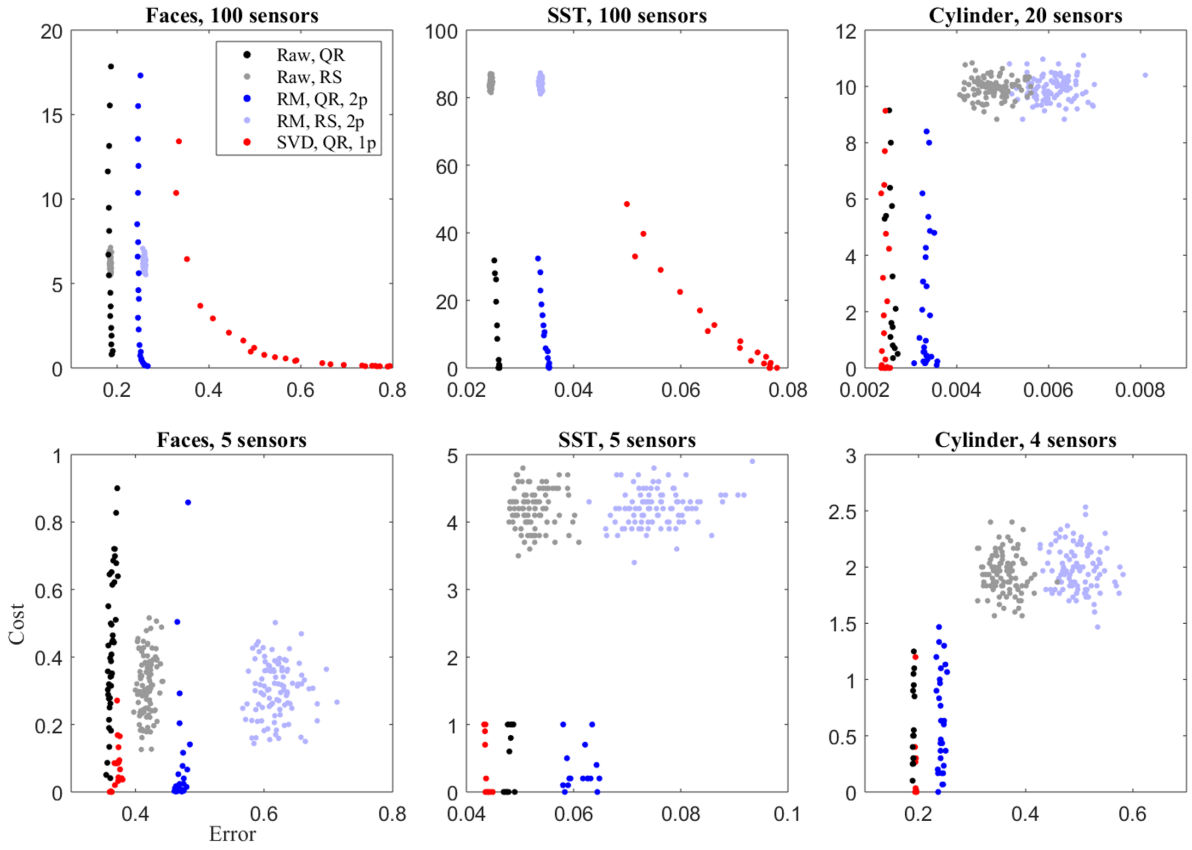

Figure 3.3: A comparison of cost versus error results with the following pre-processing methods: Taking the full raw data as a basis, $\mathbf{\Psi} = \mathbf{X}^{tr}$; taking $2p$ randomized linear combinations of $\mathbf{X}^{tr}$, where p is the number of sensors; and taking $\mathbf{\Psi}$ to be the first p SVD modes of $\mathbf{X}^{tr}$. The training set is interpolative, not extrapolative. Sensors are placed with our modified QR-based algorithm for each of the three bases and data sets, for both a comparatively large and small number of sensors. Additionally, cost and error results are calculated for randomly-selected sensors with the first two pre-processing methods. The randomized sensor results are cross validated as usual, and there are 100 data points per method. Randomized sensors using the SVD basis had significantly higher reconstruction errors than any other method, and the results are not shown here. A Gaussian cost function is used for the eigenface example. The abbreviations in the legend are explained in Table 3.1.

the performance of randomly drawn sensors for the sake of comparison. All three data sets are considered, with both a large and small number of sensors. See Table 3.1 for details on the figure labels.

For both the eigenface and sea surface temperature data sets, using the raw data at 100 sensors leads to the lowest error at a comparable cost, and randomized linear combinations with $2p$ modes gives the next lowest error, followed by SVD with p modes. At 5 sensors, the latter is reversed, with SVD performing comparably to or better than the raw data, as is the case with both trials for the fluid flow behind a cylinder. We observe that our QR-based method generally outperforms the best randomly-chosen sensors. The randomized sensors sometimes yield lower errors, particularly with a large number of sensors, but our principled method is capable of producing lower costs. We note that the randomized data performs worse than the raw data in all examples, but the behavior is consistent and may be worth the reduced computational cost in some applications.

In the remainder of this section, we more closely analyze the performance of our algorithm for each of the data sets. For brevity, we consistently use a randomized linear combination of data vectors for the pre-processing technique in these examples, noting that the behavior described above would be maintained if all techniques were tested.

3.5.1 Eigenfaces

The algorithms are first tested on the eigenface data set. All three cost functions and a few corresponding sensor arrays are shown in Figure 3.4. These are the average locations from placing 200 sensors over the twenty cross-validation runs, shown as a scatter plot on top of the dominant eigenface mode. Marker size and color indicate the frequency with which a sensor was placed at a given location, with white indicating the most frequent, shading through yellow and red to black for the least frequent. As expected, when the cost function weighting is increased, sensors are gradually pushed out of the regions of higher cost. This allows the total cost to be lowered at the expense of decreasing reconstruction accuracy, as demonstrated in the right-hand column of the figure, which plots cost and error on separate

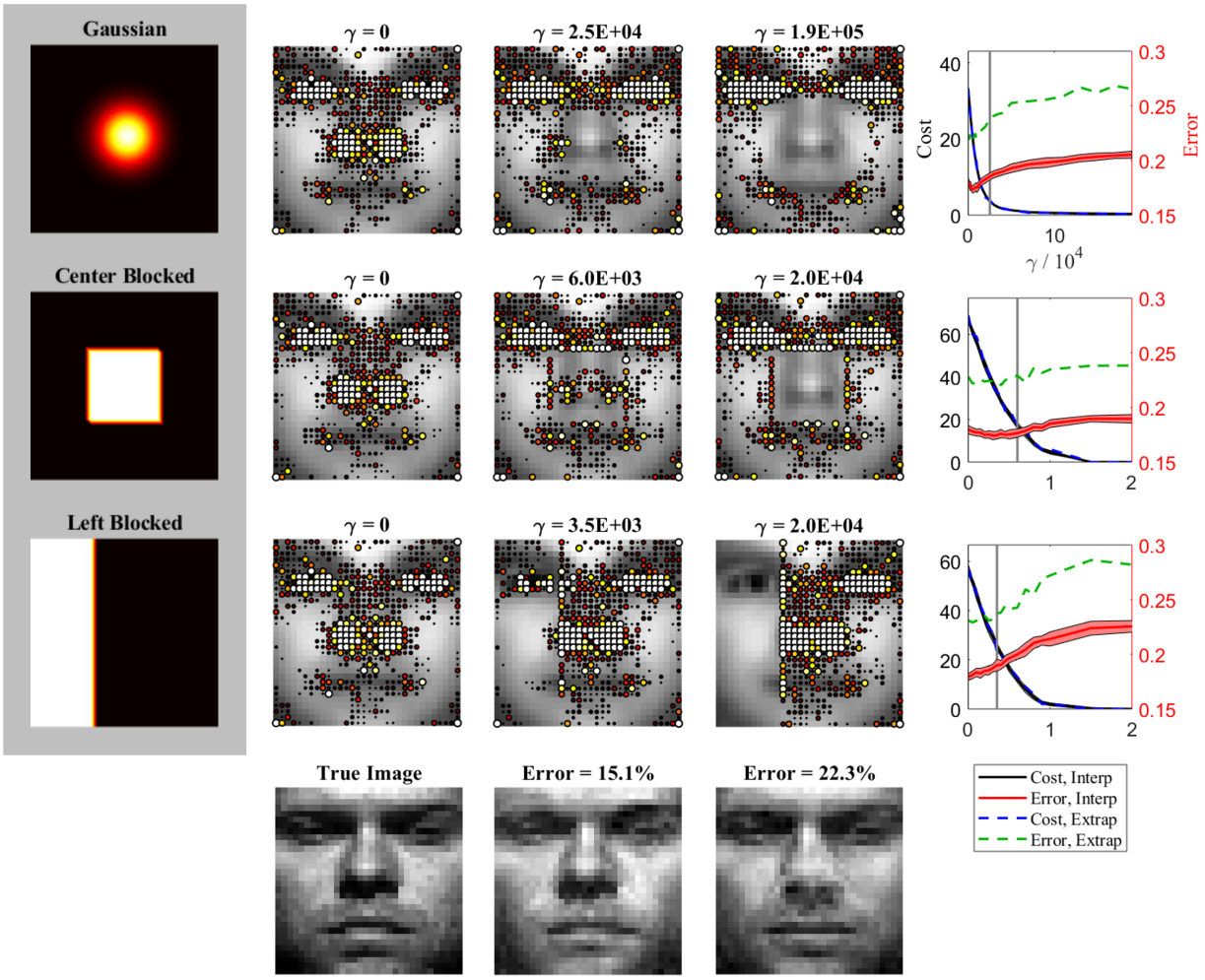

Figure 3.4: Average sensor locations for eigenface reconstruction for three different cost functions. The cost functions are plotted in the left column, where white indicates regions of highest cost and black shows regions of zero cost. The three central columns show the locations of 200 sensors placed by the QR-based algorithm, averaged over 20 cross validations, for increasing values of the weighting factor γ. The final column plots cost and error against γ for each cost function, with bands indicating error bars. Both interpolation and extrapolation results are given. The vertical gray lines indicate the value of γ at which the sensors are plotted in the middle column. The bottom row shows typical example reconstructions of one of the photos, for reference.

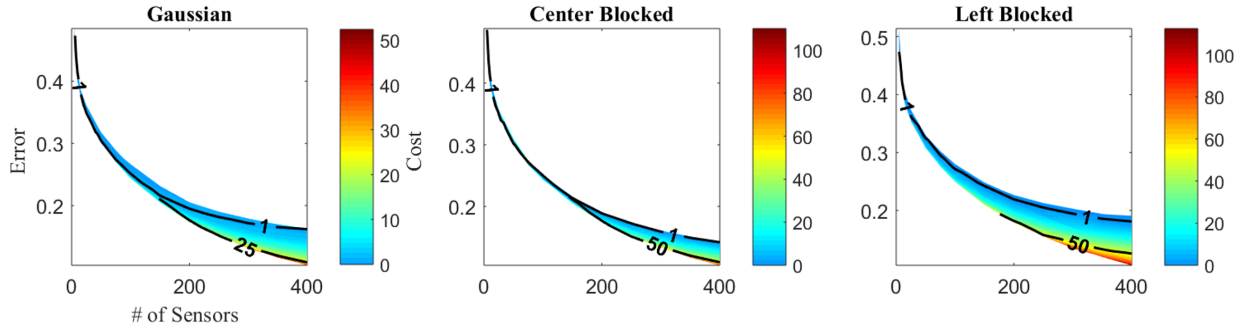

Figure 3.5: The cost landscape for eigenface reconstruction for all three cost functions, with cost plotted as a two-dimensional color map against error and the number of sensors. Contours indicate lines of constant cost.

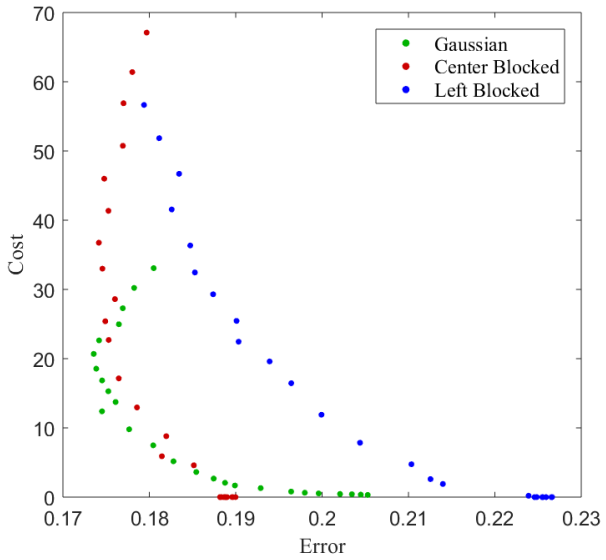

Figure 3.6: Cost versus error for eigenface reconstruction with three different cost functions. Results are given for the case of 200 sensors.

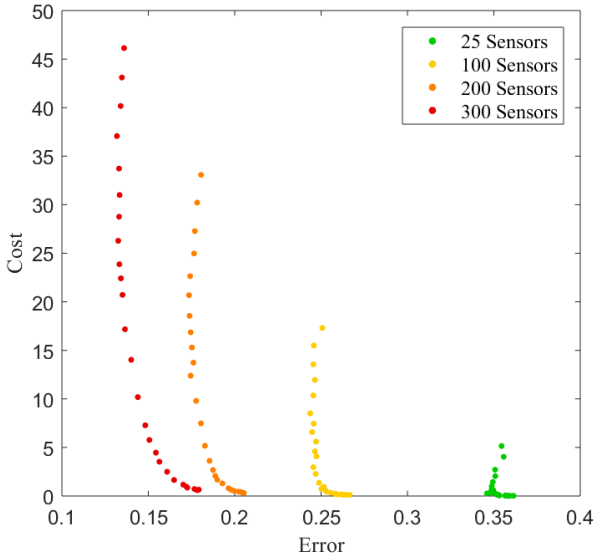

Figure 3.7: Cost versus error for eigenface reconstruction with a Gaussian cost function, for several different numbers of sensors.

axes, as a function of γ. Note that γ is an arbitrary weighting, and the same value of γ can have very different effects depending on the cost function. Extrapolative cost and error are shown in the same panel, where the extrapolative error is higher than the interpolative, at an identical cost. Additionally, the bottom row of the figure shows several reconstruction examples for one of the faces.

For many practical applications of these methods, the final goal will be to minimize reconstruction error given a predetermined budget. To that end, cost-error landscapes are constructed by calculating sensor array cost and reconstruction error for different numbers of sensors. The results are shown in Figure 3.5, which shows the landscapes for each cost function, plotted as a color map according to cost. Cost contours on this color map represent a hypothetical budget, so the optimum configuration for a given budget can be found by following the appropriate contour to the lowest possible error. Note that the upper edges of the contour plots indicate the minimum cost and maximum error for a given number of sensors, and the lower edges indicate the minimum error and maximum cost.

Cross sections of the cost landscapes as plots of cost versus error are given in Figures

3.6 and 3.7. The former shows cross sections for each of the three cost functions, using 200 sensors, and the latter shows cost versus error for a Gaussian cost function with 25, 100, 200, and 300 sensors.

3.5.2 Sea surface temperature

Next we consider the sea surface temperature data set. Average sensor locations over ten cross validations with 200 sensors are shown in Figure 3.8, as a scatter plot where the size and color of a data point indicate the frequency with which a sensor was placed in that location (blue being the least frequent, red being the most frequent). As the cost function weighting is increased, more sensors move within the unblocked regions offshore, until the cost reaches zero. Plots of cost and error as functions of γ are given in the fourth panel, and the bottom row shows two example reconstructions. As with the eigenfaces, the interpolative trial has a much lower error than the extrapolative trial. Notice that although the reconstruction error increases as cost decreases, here the error does not even reach 3%, even when the cost is zero.

The landscape of cost as a function of error and the number of sensors is shown in Figure 3.9, again visualized as a color map with contours of constant cost. Cross sections of cost versus error for 25, 100, 200, and 300 sensors are shown in Figure 3.10.

3.5.3 Fluid flow around a cylinder

Finally, we test the algorithm on periodic vortex shedding behind a stationary cylinder. This data set is low-rank, periodic, and vertically symmetric, making it a significant contrast to the previous two examples. The vertically symmetric cost function further allows the algorithm to take full advantage of the symmetry of the fluid flow. In Figure 3.11, sensor locations for several values of γ are shown. The locations are shown for 14 sensors, averaged over the 30 cross validations and then graphed as a scatter plot on top of an example fluid flow snapshot. The size and color of a data point indicate how frequently a sensor was placed at its location,

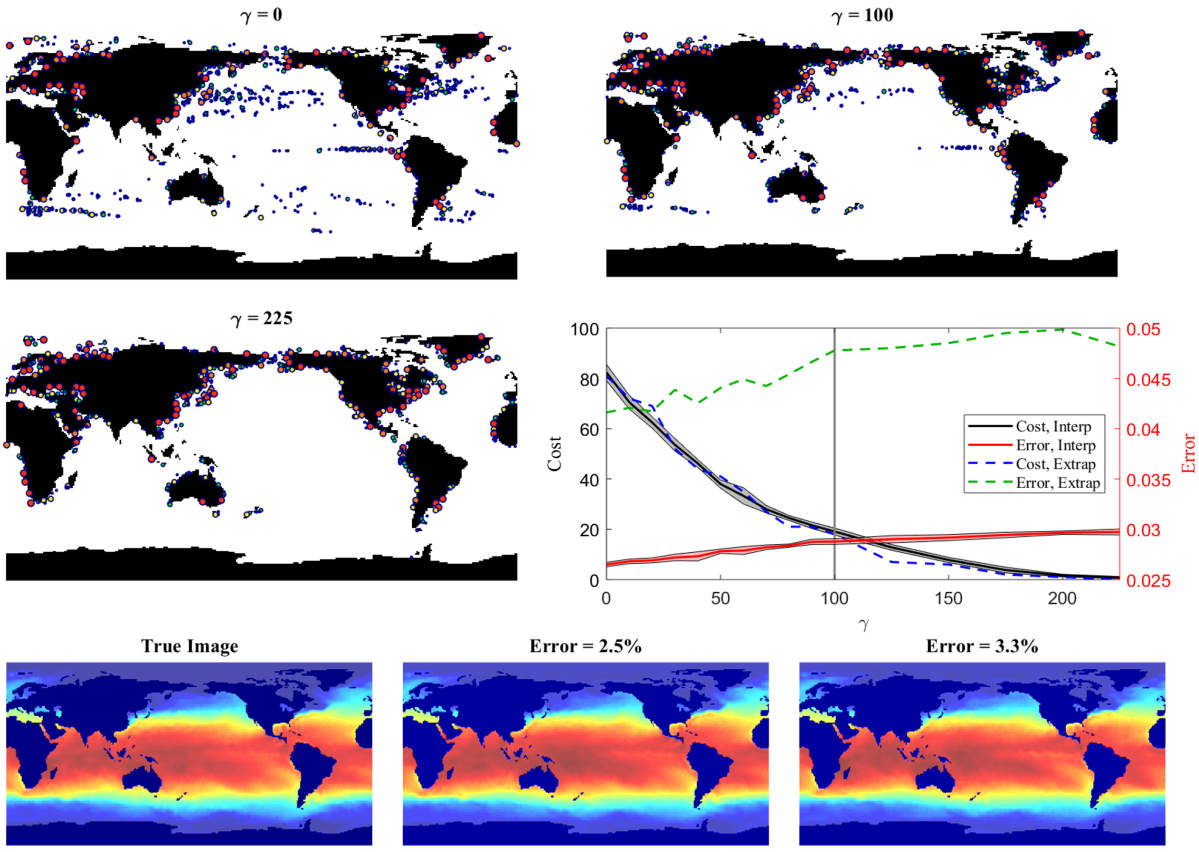

Figure 3.8: Sensor locations for sea surface temperature reconstruction with 200 sensors. The cost function considered was a step function which was zero up to two pixels off land and equal to one everywhere else. Locations are shown for three different values of the weighting factor γ, and are averaged over ten cross validations. Size and color of a data point indicate the frequency with which a sensor was placed there. The fourth image plots cost and error against γ, for both interpolative and extrapolative data sets. The bottom row shows a comparison of an example temperature snapshot along with two reconstructions of it yielding two different accuracies.

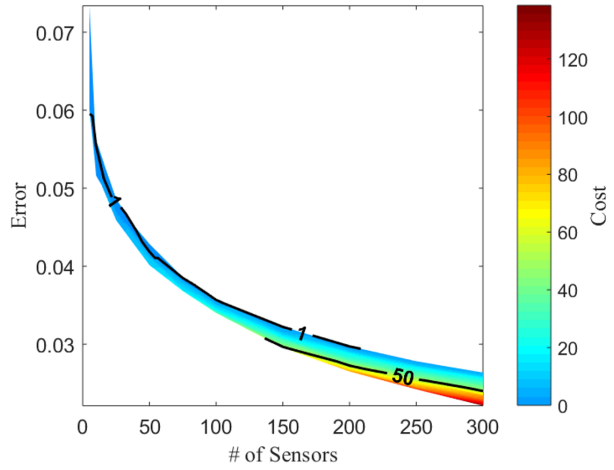

Figure 3.9: The cost landscape for sea surface temperature reconstruction, plotted as a color map against error and the number of sensors. Contours show lines of constant cost.

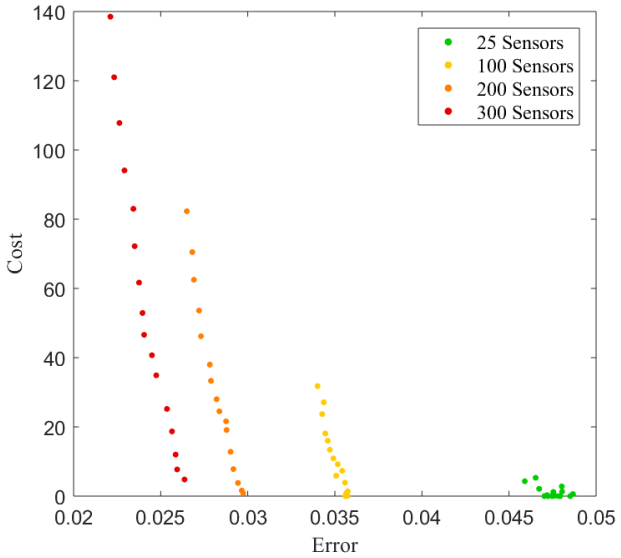

Figure 3.10: Plots of cost versus error for sea surface temperature reconstruction, with varying numbers of sensors.

with white being the most frequent and black being the least frequent. As expected, when γ is increased, the sensors migrate until they occupy the upper half of the plane.

The fourth panel of the figure plots the cost and error on separate axes, as functions of γ. Because of the symmetry of the data set, the reconstruction error is essentially unchanged with γ, even as the cost goes to zero. Furthermore, because the flow is periodic, the extrapolative data performs slightly better than the interpolative data. The figure's bottom row shows example reconstructions of a snapshot.

The fluid flow cost-error landscape is shown in Figure 3.12. Notice that there is hardly any variation in the error, so rather than plotting cost as a color map, the error versus number of sensor curves for $\gamma = 0$ (highest cost) and $\gamma = 15$ (lowest cost) are shown. These curves are essentially identical, further emphasizing that for this particular data set and cost function, the total cost can be lowered with no penalty to the error.

Cost versus error plots are given in Figure 3.13, which shows results for 4, 8, 12, and 20 sensors. Regardless of the number of sensors, these cross sections are essentially vertical lines, within the standard deviation of the error.

3.5.4 Run time for the experiments

To get a sense of how long it takes to compute sensor locations, we provide some average run time information in Table 3.2. Note that m denotes the number of training samples, n denotes the dimension of the data, p denotes the number of sensors, and t denotes the average run time in seconds[2] for a single value of γ. For all applications, p is selected to be the maximum number of sensors placed in any of the experiments above. The times are averaged over 26 runs for the Yale faces data, 5 runs for the sea surface temperature data, and 29 runs for the cylinder flow data. All times are for the raw data without preprocessing.

[2]All experiments were run on a laptop with an Intel Core i7-6600u CPU (2.60 GHz, 4 cores) and 16 Gb of RAM.

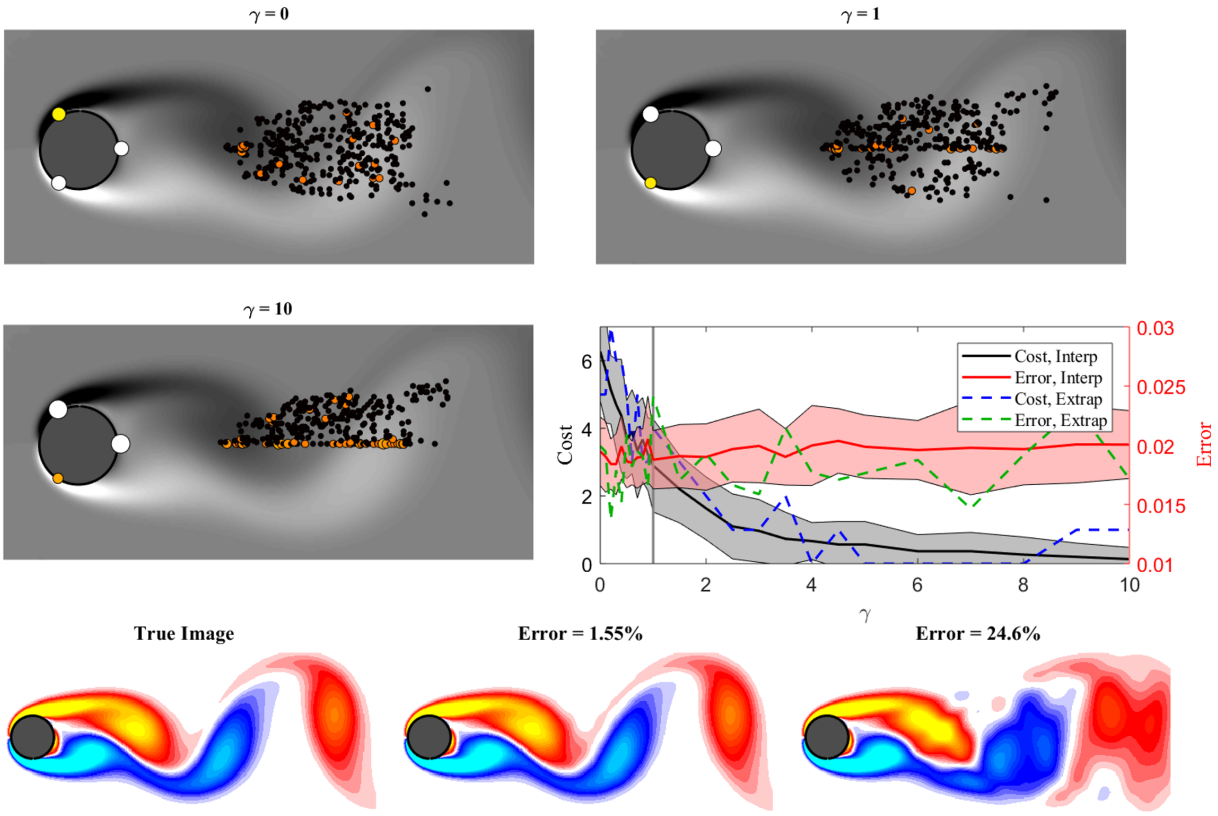

Figure 3.11: Sensor locations and reconstructions of the flow behind a cylinder, obtained by the QR-based algorithm, using 14 sensors. The cost function was a step function blocking the lower half of the domain. The first three plots show the placements of the sensors averaged over 30 trials. The fourth shows cost and error plotted against γ, where the bands indicate error bars. The bottom row gives a comparison of the true image of a snapshot along with two different reconstructions.

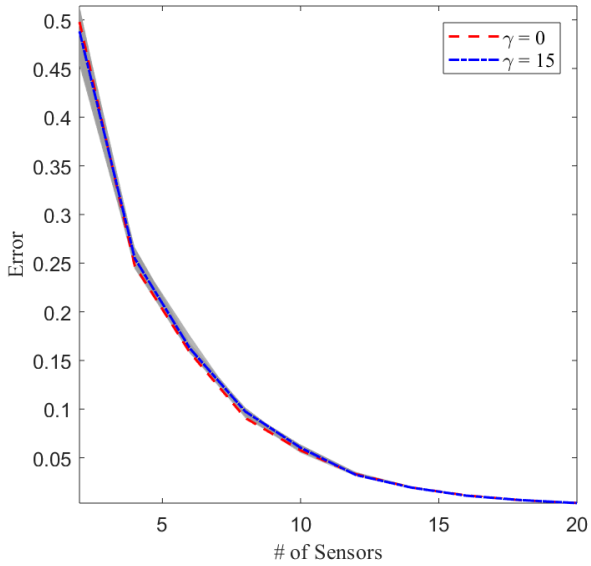

Figure 3.12: The cost landscape for reconstruction of a fluid flow behind a cylinder. Instead of the color maps made for the previous two data sets, curves of error versus the number of sensors are shown for high and low values of γ. The gray band indicates the maximum variation in the error.

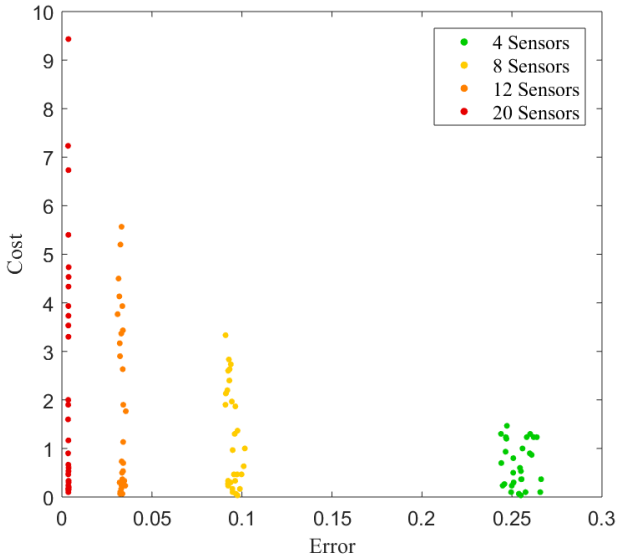

Figure 3.13: Plots of cost versus error for cylinder flow reconstruction, with varying numbers of sensors.

Data set	m	n	p	t
Yale Faces	1931	1024	400	0.82
Cylinder	129	89351	20	0.52
Sea Surface Temperature	1100	44219	300	25.65

Table 3.2: Average run time information for running Algorithm 2 on the raw data for each application.

3.6 Conclusions

We have developed a QR-based greedy algorithm to place sensors for reconstruction with a cost constraint on sensor locations. This algorithm is tested on three different data sets, eigenfaces, weekly sea surface temperature data, and vortex shedding of a fluid flowing around a cylinder. In all cases, the method is demonstrated to be capable of lowering sensor cost at the expense of marginal increases in reconstruction error.

The algorithm presented here provides a way to place sensors under a cost constraint, which could have applications in manufacturing, atmospheric sensing, fluid flow sensing, and many more fields. Specifically, the algorithm allows one to address three critical engineering design principles in regard to sensors selection: (i) For a fixed budget of sensors, where are the best measurement locations, (ii) What is the minimal number of sensors required to achieve a given reconstruction error, and (iii) How well can inaccessible regions be reconstructed in practice. Depending upon the application, one or all of these questions may be of central concern. The computationally tractable approach presented here provides a principled mathematical method for answering these questions.

Chapter 4

OVER- AND UNDERSAMPLING FOR THE SVD AND RANDOMIZED BASES

In Chapter 3, we noticed that using a basis of SVD modes with an equal number of sensors p and modes r caused the reconstruction error to unexpectedly increase with the number of sensors. This effect was also observed in, e.g. [79], and in Chapter 3 we avoided it by using a randomized rank reduced basis with $r = 2p$ as suggested in, e.g. [37,38]. However, Peherstorfer et al [77] proved that with an SVD basis and $p = r$, error increases as $\sqrt{p}$ due to the amplification of noise in the system when the reconstruction is performed. They showed that this can be avoided by using more sensors than SVD modes, which leads to lower reconstruction errors that decrease with p. We now perform a systematic comparison of the two rank reduction and sampling methods: SVD with oversampling ($p > r$), and randomized rank reduction with undersampling ($r > p$). We calculate the reconstruction error as a function of r and p for the Yale B Face data set with both bases, and determine the minimum possible error at a given number of sensors, and the number of modes at which this minimum error occurs. We find that the two methods have nearly identical minimum errors, and that for the SVD basis, this minimum error occurs at approximately $p = 1.4r$ oversampling, while for the randomized basis, the error is nearly always minimized with $r = 800$ modes (the maximum number of modes we tested), independent of the number of sensors. We also compare two oversampling techniques, randomized oversampling and principled oversampling through eigenvalue tuning. We find that for the eigenface data set, the slight improvement in reconstruction error by principled oversampling is probably not worth the increased runtime to perform it.

Section 4.1, briefly presents sparse sensing with the SVD and randomized bases, and

justifies over- and undersampling, respectively. In Section 4.2 we perform a direct comparison between the two bases with varying degrees of over- and undersampling.

4.1 Introduction

In this section, we briefly summarize theoretical results regarding oversampling in an SVD basis and undersampling with a randomized basis.

4.1.1 SVD with oversampling

The first basis considered here consists of the first r left singular vectors of $\mathbf{X}^{tr}$:

$$\mathbf{X}^{tr} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^* \tag{4.1}$$

$$\mathbf{\Psi} = \mathbf{U}_{:,1:r}, \tag{4.2}$$

This is the default choice for many sparse sensing applications. The SVD provides the optimal low-rank approximation of a data set [59], and the SVD basis has proven highly effective for sparse sensing, see, for example, [25, 36]. However, Peherstorfer, Drmač, and Gugercin [77] discover that the convention of taking the same number of modes and sensors in an SVD basis leads to noise amplification, with the result that reconstruction error actually increases with p.

From [77], Lemma 3.1: Let there be noise present in the system, such that the state of the test set with noise is given by

$$\mathbf{X}^{te}_{\boldsymbol{\epsilon}} = \mathbf{X}^{te} + \boldsymbol{\epsilon}, \tag{4.3}$$

where $\boldsymbol{\epsilon} \in \mathbb{R}^{n\times\ell}$ has Gaussian i.i.d. entries with standard deviations $\boldsymbol{\sigma} \in \mathbb{R}^n$. We take an SVD basis with $p = r$. The expectation over noise of the reconstruction error is

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left(||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Theta}^{\dagger}_J\mathbf{C}_J\mathbf{X}^{te}_{\boldsymbol{\epsilon}}||_2\right) \leq ||\mathbf{\Theta}^{\dagger}_J||_2\left(||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Psi}^*\mathbf{X}^{te}||_2 + \sqrt{p}||\mathbf{C}_J\boldsymbol{\sigma}||_\infty\right). \tag{4.4}$$

The term due to the projection error, $||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Psi}^*\mathbf{X}^{te}||_2$, decreases with r [59], while the noise term, $\sqrt{p}||\mathbf{C}_J\boldsymbol{\sigma}||_\infty$, increases as $\sqrt{p}$, quickly dominating the reconstruction.

This can be mitigated through oversampling, taking $p > r$.

From [77], define the coherence of a subspace $\mathcal{U} = \text{span}(\mathbf{\Psi})$, for $\mathbf{\Psi}$ an orthonormal basis of $\mathcal{U}$, as

$$\mu(\mathcal{U}) = \frac{n}{r} \max_{i=1,\dots,n} ||\boldsymbol{\psi}_i^T||_2, \tag{4.5}$$

where $\boldsymbol{\psi}_i^T$ is the i^{th} row of $\mathbf{\Psi}$.

Let the set of sensor locations K be randomly drawn with replacement from $\{1, \dots, n\}$, and let $\mathbf{C}_K$ be the corresponding selection matrix. Futhermore, let $\delta > 0$, $p \geq (8/3)r\mu(\mathcal{U})\log(2r/\delta)$, and $\gamma = \sqrt{\frac{8r\mu(\mathcal{U})}{3p}\log(2r/\delta)}$. Finally, define

$$\xi = \min\left\{\frac{1}{\sqrt{1-\gamma}}, \frac{1}{\sqrt{1-\gamma}}\sqrt{\frac{r}{p}\mu(\mathcal{U})}\right\}. \tag{4.6}$$

Then, from [77], Theorem 4.6:

$$\mathbb{E}_{C_K}\left[\mathbb{E}_{\boldsymbol{\epsilon}}\left(||\mathbf{X}^{te} - \hat{\mathbf{X}}_{\boldsymbol{\epsilon}}^{te}||_2\right)\right] \leq (1+\xi)||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Psi}^*\mathbf{X}^{te}||_2 + \frac{||\boldsymbol{\sigma}||_\infty}{(1-\gamma)}\sqrt{\frac{rn}{p}}, \tag{4.7}$$

with probability at least $1 - \delta$, where $\mathbb{E}_{C_K}$ is the expectation over random sensor selections. Thus as $p \to \infty$, the upper bound on the error expectation converges to the projection error bound, $||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Psi}^*\mathbf{X}^{te}||_2$.

Thus, even with random sensors, oversampling negates noise amplification. Peherstorfer et al also present several principled oversampling techniques, one of which we will consider below: principled oversampling through eigenvalue tuning. The method places sensors so as to maximize the smallest eigenvalue of $\mathbf{\Theta}_J$, which should provide highly stable reconstructions. This is accomplished by a greedy algorithm that at every step $k = r+1, \dots, p$, adds the row of $\mathbf{\Psi}$ that has maximum inner product with the k^{th} right singular vector of $\mathbf{\Theta}_{J^{k-1}}$. This procedure is outlined in Algorithm 3, see [77, 80] for a more complete treatment of interpolant selection via eigenvalue tuning.

Algorithm 3 Oversampling through eigenvalue tuning, ODEIM+E, from [77].

Input: Matrix $\mathbf{\Psi} \in \mathbb{R}^{n \times r}$, QR sensors J, $|J| = r$.
Output: ODEIM+E sensors, J, $|J| = p$.

for $k = r+1, \ldots, p$ **do**
  $[\sim, \sim, \mathbf{W}] \leftarrow \texttt{SVD}(\mathbf{\Psi}_{J,:})$
  for j=1,. . . ,n **do**
    $\boldsymbol{\phi}(j) \leftarrow ||\mathbf{W}^T_{:,r}\mathbf{\Psi}^T_{:,j}||_2$
  $[\sim, I] \leftarrow \texttt{sort}(\boldsymbol{\phi}, \text{'descend'})$
  $i \leftarrow 1$
  while any$(I(i) = J)$ **do**
    $i \leftarrow i + 1$
  $J(k) \leftarrow I(i)$
return J

4.1.2 Randomized basis with undersampling

The second basis we consider is randomized linear combinations of the columns of $\mathbf{X}^{tr}$, as described in, e.g., [37, 38, 81]:

$$\mathbf{\Psi} = \mathbf{X}^{tr}\mathbf{G}, \tag{4.8}$$

where $\mathbf{G} \in \mathbb{R}^{m \times r}$ has i.i.d. Gaussian random entries. This basis is faster to calculate than the SVD, making it particularly useful for large data sets, and it is able to capture much of the system's energy with high statistical likelihood. The likelihood of randomized rank reduction sufficiently capturing the system's energy increases with the number of modes, where the number of modes should be greater than the rank of the system.

From [38], Theorem 1.1: Select a target rank r_t, and an oversampling parameter $q \geq 2$, where $r_t + q \leq \min\{n, m\}$, where $\mathbf{X}^{te} \in \mathbb{R}^{n \times m}$. Then, with randomized basis $\mathbf{\Psi} = \mathbf{X}^{te}\mathbf{G}$, where $\mathbf{\Psi} \in \mathbb{R}^{n \times (r_t+q)}$,

$$\mathbb{E}_{\mathbf{G}}\left(||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Psi}^{\dagger}\mathbf{X}^{te}||_2\right) \leq \left[1 + \frac{4\sqrt{r_t+q}}{q-1}\sqrt{\min\{n,m\}}\right]\sigma_{r_t+1}, \tag{4.9}$$

where σ_{r_t+1} is the $(r_t+1)^{th}$ singular value of $\mathbf{X}^{te}$.

In fact,

$$||\mathbf{X}^{te} - \mathbf{\Psi}\mathbf{\Psi}^{\dagger}\mathbf{X}^{te}||_2 \leq \left[1 + 9\sqrt{r_t + q}\sqrt{\min\{n, m\}}\right] \sigma_{r_t+1}, \tag{4.10}$$

with probability at least $1 - 3q^{-q}$, implying that only small values of q are necessary, and that for large values of q, the error from the randomized basis is within a polynomial factor of the theoretical minimum of σ_{r_t+1} with high probability. We note that these results are actually presented for an orthogonal basis which is initialized as in Equation 4.8, but whose columns are then orthogonalized through a Graham-Schmidt procedure. However, the results generalize to a non-orthogonal basis, and our sparse sensing setup and QR algorithm do not require an orthogonal basis.

Because $\mathbf{X}^{te}$ is downsampled through sparse sensing, here the target rank is the number of sensors, $r_t = p$. Thus, to construct a randomized basis, we should take $r = p + q$, for $q \geq 5$.

4.2 Effects of the number of sensors and modes

We now present a thorough exploration of the effect of the number of sensors and modes on reconstruction error for the two bases described above. We vary the number of modes and sensors to determine the optimal amount of over- and undersampling for each basis and which basis has the best performance overall. The tests are performed on the Extended Yale Face Database B [66–69], as described in Section 3.3. Recall that the data has a slow singular value decay with a Gavish-Donoho cutoff [76] around rank 350. We randomly select 80% of the snapshots for $\mathbf{X}^{tr}$ and the rest of the photographs form the test set $\mathbf{X}^{te}$.

We select the number of modes r, the number of sensors p, and a basis, either Equation 4.2 or 4.8, then use Algorithm 1 to select sensors in the case of $p \leq r$. When $p > r$, unless otherwise stated randomized oversampling is performed, that is, the first r sensors are the set J of QR pivots, and the remaining $p - r$ sensors are randomly drawn from the set of unselected locations, $\{1, \ldots, n\} \backslash J$.

We let the sensors have 2% signal-to-noise levels, such that the measurements are given

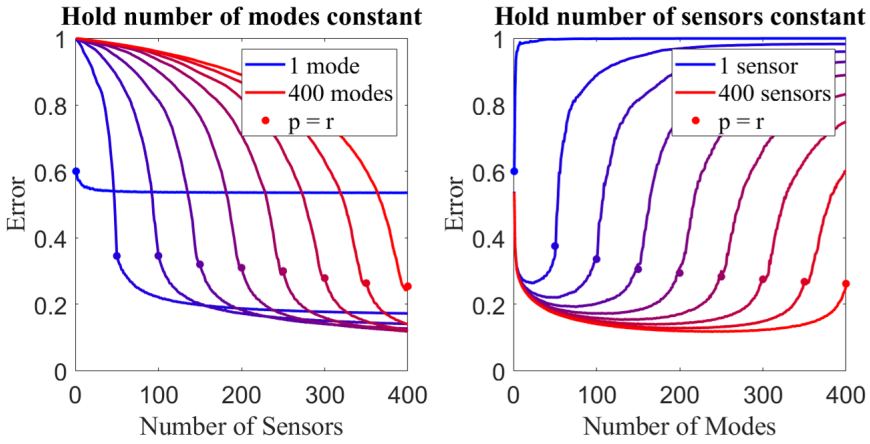

Figure 4.1: Reconstruction error for the eigenface data set, taking a basis of SVD modes. The left plot shows error versus the number of sensors as the number of modes is held constant, while the right plot gives error versus the number of modes, holding the number of sensors constant. The blue line indicates one mode (sensor), shading through to red with 400 modes (sensors) in increments of 50. The dots show the point at which the number of sensors is equal to the number of modes. Randomized oversampling is performed where $p > r$.

by

$$\mathbf{Y}(J) = \mathbf{C}_J \mathbf{X}^{te} + \boldsymbol{\epsilon}, \tag{4.11}$$

where $\boldsymbol{\epsilon} \in \mathbb{R}^{p \times \ell}$ has Gaussian i.i.d. entries with variance 2% of the overall variance of the data set. We perform reconstruction as in Equation 2.7 and evaluate the sensor performance by the fractional reconstruction error, Equation 2.9b. The results are averaged over 20 random training and test sets, each with 10 realizations of random noise.

Figure 4.1 and 4.2 show results of varying the number of modes and sensors, with an SVD basis for the former figure and a randomized basis for the latter. Both figures have the same layout. The left plot shows reconstruction error versus the number of sensors, where each line has a constant number of modes, while on the right, error is plotted as a function of the number of modes, where each line has a constant number of sensors. The blue lines have one mode (sensor), and the colors shade through to red with 400 modes (sensors). For the randomized basis, we continue through to 800 modes, shown in yellow.

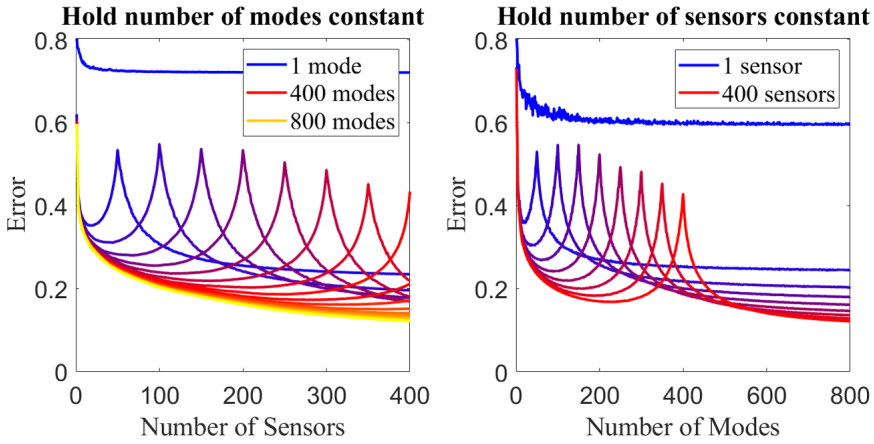

Figure 4.2: Reconstruction error for the eigenface data set, taking a basis of randomized modes, with randomized oversampling in the case of more sensors than modes. The left plot shows error versus the number of sensors as the number of modes is held constant, while the right plot gives error versus the number of modes, holding the number of sensors constant. The blue line indicates one mode (sensor), shading through to red with 400 modes (sensors), and on the left plot, continuing on to yellow with 800 modes.

Figure 4.1 makes it clear that oversampling is indeed preferred in an SVD basis. On the left plot, the error continues to decrease as sensors are added beyond the number of modes, and equivalently on the right plot there are minima when $r < p$. Meanwhile, the lines in Figure 4.2 have pronounced peaks at $p = r$, suggesting that the randomized basis performs well with either over- or undersampling, but very poorly when the number of modes and sensors are very similar.

We also test principled oversampling through ODEIM+E. This is significantly slower to perform than randomized oversampling, and in this case it performs only slightly better. This can be seen in Figure 4.3, which shows reconstruction error versus p, with 1, 100, 200, and 300 modes. Results are shown for randomized modes in the upper plot and SVD modes in the lower, with the solid line showing randomized oversampling and the dashed line showing ODEIM+E. Beyond $r = 1$, principled oversampling outperforms ODEIM+rand by only about 1% for both basis choices, and given the increased runtime to perform ODEIM+E, demonstrated in Figure 4.4, we determine that for the eigenface data set, it makes sense

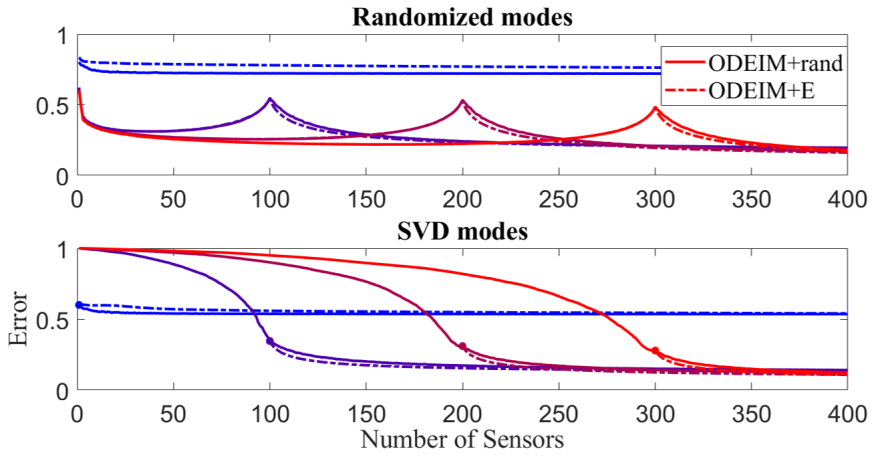

Figure 4.3: A comparison of randomized oversampling ODEIM+rand, with principled oversampling through eigenvalue tuning, ODEIM+E. In both subplots, error versus the number of sensors is plotted as the number of modes is held constant, with blue showing one mode and red showing 300 modes. Where $p > r$, randomized oversampling is shown as a solid line and principled oversampling is shown as a dashed line. The upper plot shows results with a randomized basis, and the lower plot gives results with an SVD basis.

to use randomized oversampling. The right subplot of Figure 4.4 shows the CPU time to place the last $p - r$ sensors in an SVD basis using both methods, plotted as a function of the number of sensors. Each line shows time versus p for a constant number of modes, and it is apparent that as the number of modes increases, the time to perform randomized oversampling remains approximately constant as the number of modes increases, while the time to perform ODEIM+E increases by up to two orders of magnitude.

Most importantly for this section, we must determine which basis provides the lowest reconstruction error. In Figure 4.5, we consider both bases and determine the lowest error at a given number of sensors, plotted on the left, and the number of modes at which this minimum error occurs, plotted on the right. We also consider the local minima present in the randomized mode plots by calculating the minimum error with $r < p$. The results indicate that the randomized and SVD bases have nearly identical minimum errors, but while the SVD basis prefers oversampling, the randomized basis requires close to the full 800 modes. When the randomized basis is restricted to oversampling, the minimum error is up to 10%

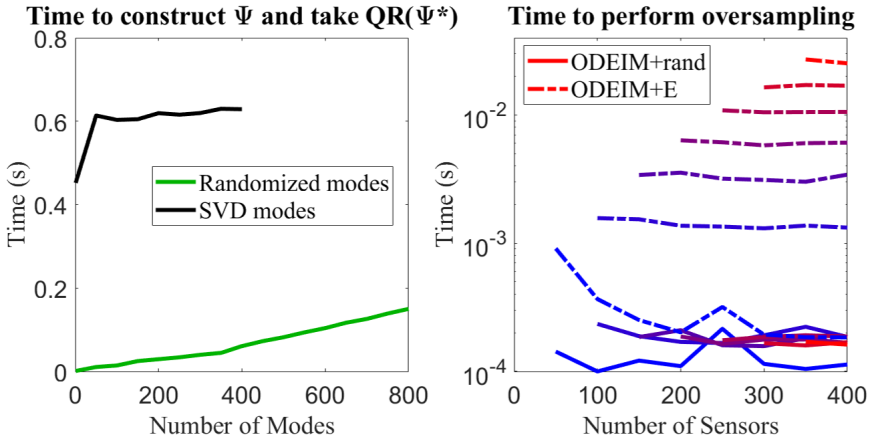

Figure 4.4: Time comparisons for basis choice (left) and oversampling method (right). For basis choice, we consider the combined time to construct the randomized and SVD bases and then to perform the column-pivoted QR decomposition. On the right, we calculate the time to perform randomized oversampling (solid lines) and principled eigenvalue-based oversampling (dashed lines), with an SVD basis. We use 1 to 300 modes and up to 400 sensors, with 1 mode plotted in blue and 300 modes in red. Both r and p increase in increments of 50.

higher than with undersampling or taking the SVD basis. The randomized basis is faster to construct than the SVD basis, as shown on the left plot of Figure 4.4, which plots the CPU time to construct $\mathbf{\Psi}$ and perform the column-pivoted QR decomposition, as a function of the number of modes for both bases. Thus the randomized basis is appealing, particularly for a very large data set where the speed-up is highly desirable. However, in many sensor placement examples, one of the goals is to perform rank reduction, a task for which the SVD is better suited, and so a researcher should carefully consider their data set and their sparse sensing goal before selecting a basis.

Note that fitting a linear regression to $p(r)$ for the SVD data yields a slope of about $p = 1.4r$. (For the randomized modes with oversampling, $p = 1.6r$.) In this text, we take $p = 2r$ when we use an SVD basis, to ensure that the degree of oversampling is identical for any number of sensors, since sensors and modes must be integers. Similarly, if we take a randomized basis, we use $r = 2p$ modes, to achieve a balance between performance and rank

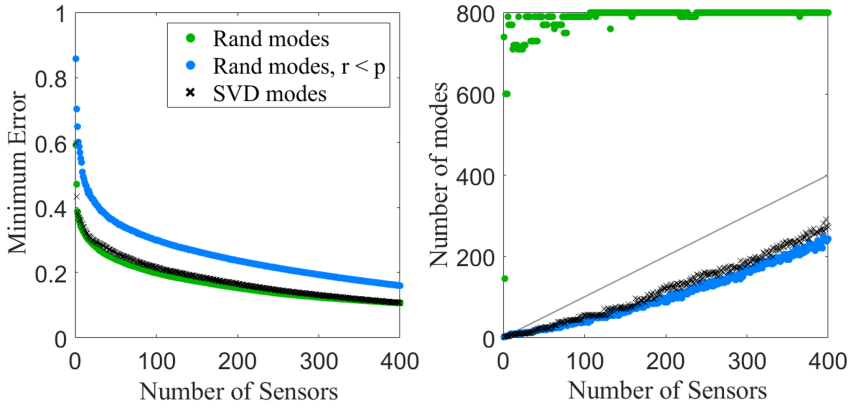

Figure 4.5: On the left is plotted the minimum possible error at a given number of sensors, for a basis of randomized modes in green, SVD modes in black, and randomized modes restricted to fewer modes than sensors in blue. The right panel shows the number of modes at which the minimum error occurs, also as a function of the number of sensors, in the same color scheme as that on the left. Also included for reference is the line $r = p$. No noise is incorporated here, to save on runtime.

reduction.

4.3 Conclusions

We test the effect of over- and under-sampling given one of two common basis choices. With an SVD basis, oversampling stabilizes the reconstruction, while with a randomized reduced basis, taking more modes than sensors captures more of the system's energy and leads to improved results. We find that for the Yale B Face data set, the SVD with oversampling yields nearly identical reconstructions to randomized modes with under-sampling, but while the SVD performs best with only moderate oversampling, the randomized basis results improve continually as more modes are added. In most applications, sparse sensor placement is performed alongside rank reduction, so the SVD will probably be the preferred choice. However, randomized rank reduction is faster to perform, so it may be preferable for a very large system.

Chapter 5

SENSOR SELECTION WITH COST CONSTRAINTS FOR DYNAMICALLY RELEVANT BASES

We apply the cost-modified column-pivoted QR decomposition to dynamical systems for which the usual singular value decomposition (SVD) basis may not be available or preferred. In considering different bases, we are able to account for the dynamics of the particular system, yielding sensor arrays that are nearly Pareto optimal in sensor cost and performance in the chosen basis. This flexibility extends our framework to include actuation and dynamic estimation, and to select sensors without training data. We provide three examples from the physical and engineering sciences and evaluate sensor selection in three dynamically relevant bases: truncated balanced modes for control systems, dynamic mode decomposition (DMD) modes, and a basis of analytic modes. We find that these bases all yield effective sensor arrays and reconstructions for their respective systems. When possible, we compare to results using an SVD basis and evaluate tradeoffs between methods.

5.1 Introduction

Modal decompositions and model reduction are fundamental to many fields of science and engineering [82–84]. Historically, such decompositions were based on special functions, such as Fourier modes, Legendre polynomials, and Bessel functions. The development in the late 1960s [85] of an efficient numerical procedure for computing the singular value decomposition (SVD) allowed for data-driven modal decompositions that far outperformed generalized functional techniques. The SVD is now the dominant paradigm for model reduction in many applications, and for good reason—for a given number of modes, it is guaranteed to capture the greatest amount of the system's energy (variance) [59]. However, ordering modes based

on energy does not result in the most effective basis for every task. For control systems, modes obtained via balanced truncation [86] are often more effective than SVD modes. Similarly, the dynamic mode decomposition (DMD) [87–90] provides an alternative basis that simultaneously captures spatial correlation and evolution in time. Sparse sensor optimization is often based on SVD modes [25,78], although this approach has been previously extended to include these alternative bases [91,92]. The goal of this chapter is to apply the cost-modified column-pivoted QR decomposition [78] to these alternative bases. This extension of the method increases the practicality of sensor selection by both accounting for sensor cost and using a basis that is matched to the task at hand (i.e., reconstruction, estimation, or control). First, we demonstrate balanced sensor and actuator selection for control systems, following the method of Manohar et al. [92], which we modify to include a cost function. Next, we perform sensor selection on a dynamic mode decomposition (DMD) basis [87]. The DMD gives approximate equations of motion for the system, allowing us to construct a Kalman filter that outperforms interpolative reconstructions in the presence of noise. Finally, we apply our method to universal basis modes, such as Fourier modes, Bessel functions, or a polynomial basis. This is useful for systems that admit a universal basis or for which the equations of motion are known and solvable for a modal decomposition. In this way, it is possible to perform principled sensor selection without training data.

In all three applications, we show that the modified column-pivoted QR decomposition yields sensor (and actuator) arrays with accurate reconstructions, while accounting for costs on sensor (and actuator) location. For the DMD and analytic mode examples, we also provide comparisons with sensors selected using an SVD basis. We find that the SVD basis and sensors often lead to better reconstructions, but the dynamically relevant bases have other advantages—information about the dynamics in the case of DMD modes, and no need for high-fidelity training data with analytically calculated modes. Along with the standard SVD basis, the balanced, DMD, and analytic bases span a wide range of possible cost-constrained sensor selection applications, from fields like industry, fluid dynamics, and experimental physics. Presumably this algorithm can be applied to other modal decompositions as well,

meaning that a researcher is free to choose the most relevant basis for his or her system and task.

The remainder of the chapter is organized as follows: In Section 5.2, we describe sensor and actuator selection for control systems, including an example from physics. Section 5.3 considers cost-constrained sensor selection for DMD modes, with an example from climate science. Section 5.4 explores sensor selection with cost for universal basis modes, demonstrated on another example from physics. We present conclusions in Section 5.5.

5.2 Balanced sensor and actuator selection for control systems

There is a large body of work regarding optimal sensor and actuator selection for control. Some, including [93,94], pose sensor and actuator selection as a convex optimization problem. Many take advantage of submodular or supermodular performance metrics to design greedy algorithms, for example [55,95–97]. Nestorović and Trajkov [98] design a sensor and actuator selection method for flexible structures such as cantilever beams based on balanced truncation and the H_2 and H_∞ norms, which is similar to the framework considered here. We follow the method of Manohar et al. [92] and perform the cost-constrained column-pivoted QR decomposition on truncated balanced modes for sensor and actuator selection. An outline of this method follows, [84, 92] contain full details.

Consider the control system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{5.1a}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x}, \tag{5.1b}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the system state, $\mathbf{A} \in \mathbb{R}^{n\times n}$ gives the dynamics without actuation, $\mathbf{u} \in \mathbb{R}^q$ is the actuation or input, $\mathbf{B} \in \mathbb{R}^{n\times q}$ describes the effect of the input on the state, and, as before, $\mathbf{y} \in \mathbb{R}^p$ is the measurements, and $\mathbf{C} \in \mathbb{R}^{p\times n}$ is the selection matrix.

We seek a suitable basis to place sensors (select rows of the identity matrix to form $\mathbf{C}$) and actuators (select columns of the identity matrix to form $\mathbf{B}$). We employ balanced

truncation [86,99,100], which uses the controllability and observability Gramians to construct a basis in which the system is simultaneously maximally controllable and observable.

The controllability and observability Gramians $\mathbf{W}_c$ and $\mathbf{W}_o$ are given by

$$\mathbf{W}_c = \int_0^\infty e^{\mathbf{A}t}\mathbf{B}\mathbf{B}^* e^{\mathbf{A}^* t} dt \tag{5.2}$$

$$\mathbf{W}_o = \int_0^\infty e^{\mathbf{A}^* t}\mathbf{C}^*\mathbf{C} e^{\mathbf{A}t} dt. \tag{5.3}$$

Their eigenvectors define the directions in which the system is most controllable and observable, ranked by the magnitude of the corresponding eigenvalues. The balancing transformation makes the controllability and observability Gramians equal to each other and diagonal:

$$\tilde{\mathbf{W}}_c = \tilde{\mathbf{W}}_o = \mathbf{\Sigma}, \tag{5.4}$$

where

$$\tilde{\mathbf{W}}_c = \mathbf{\Phi}^* \mathbf{W}_c \mathbf{\Phi}, \tag{5.5}$$

$$\tilde{\mathbf{W}}_o = \mathbf{\Psi}^* \mathbf{W}_o \mathbf{\Psi}, \tag{5.6}$$

$\mathbf{\Phi}^* = \mathbf{\Psi}^{-1}$, and $\mathbf{\Sigma}$ is diagonal. The transformation matrix can be calculated by solving an eigenvalue problem:

$$\mathbf{W}_c \mathbf{W}_o \mathbf{\Psi} = \mathbf{\Psi}\mathbf{\Sigma}^2. \tag{5.7}$$

With the eigenvalues ranked in decreasing order, the first r modes of $\mathbf{\Psi}$ and $\mathbf{\Phi}$ define the r most jointly observable and controllable directions for the system.

These modes are used to perform a rank-reduced change of basis, such that the new

control system is written as:

$$\dot{\mathbf{a}}_r = \mathbf{\Phi}_r^* \mathbf{A} \mathbf{\Psi}_r \mathbf{a}_r + \mathbf{\Phi}_r^* \mathbf{B} \mathbf{u} \tag{5.8a}$$

$$\mathbf{y} = \mathbf{C} \mathbf{\Psi}_r \mathbf{a}_r, \tag{5.8b}$$

where $\mathbf{\Psi}_r$ and $\mathbf{\Phi}_r$ are the first r columns of $\mathbf{\Psi}$ and $\mathbf{\Phi}$.

To select sensors and actuators with a cost function, we initialize $\mathbf{B}$ and $\mathbf{C}$ to identity matrices, calculate the Gramians and the truncated balancing modes, and perform the cost-constrained column-pivoted QR decomposition of Algorithm 2 on $\mathbf{\Psi}_r^*$ for sensor selection and on $\mathbf{\Phi}_r^*$ for actuator selection. We note that this method can be applied to either open or closed loop systems, and refer the reader to [92] for a more in-depth discussion of balanced sensors and actuators for open and closed loop control.

As a performance metric, take the H_2 norm,

$$\operatorname{tr}\left(\mathbf{C} \mathbf{W}_c \mathbf{C}^*\right) = \operatorname{tr}\left(\mathbf{B}^* \mathbf{W}_o \mathbf{B}\right), \tag{5.9}$$

which is a system's output energy, or the root mean square of its impulse response. A small output energy indicates that the sensor and actuator arrays are able to quickly stabilize the system after an input is applied, and therefore we wish to minimize the H_2 norm. However, we calculate a proxy:

$$\log\det \mathbf{C} \mathbf{W}_c \mathbf{C}^* = \log\det \mathbf{B}^* \mathbf{W}_o \mathbf{B}. \tag{5.10}$$

As shown in Summers [96], maximizing Equation 5.10 is equivalent to minimizing the H_2 norm. Furthermore, Manohar [92] demonstrates that $\arg\max_{\mathbf{C}} \log\det \mathbf{C} \mathbf{W}_c \mathbf{C}^*$ simplifies to $\arg\max_{\mathbf{C}} |\det \mathbf{C} \mathbf{\Psi}_r|$, hence why the greedy determinant maximization of the QR algorithm is desirable.

Our modification of adding a cost function during sensor and actuator selection removes any performance guarantees, but as will be shown in the next section, it is possible to

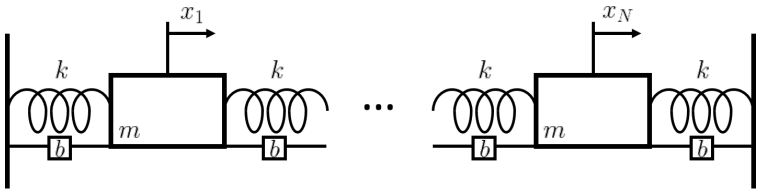

Figure 5.1: A one-dimensional system of N identical masses with mass m, connected by springs with spring constant k, and damped by a factor of b.

effectively select sensors and actuators while accounting for a nonuniform cost on location.

5.2.1 Example

We demonstrate balanced sensor and actuator selection on a system of N identical masses with mass m, connected by identical damped springs, with spring constant k and damping b [93], as illustrated in Figure 5.1. This is a simple benchmark system, and by choosing $N = 16$ masses, it is possible to compare the principled sensor and actuator selections to a brute force search over every possible sensor/actuator configuration.

The equation of motion for the i^{th} mass is given by

$$m\ddot{x}_i = k(x_{i-1} + x_{i+1} - 2x_i) + b(\dot{x}_{i-1} + \dot{x}_{i+1} - 2\dot{x}_i) + f_i,$$

where x_i is the displacement of the i^{th} mass from equilibrium and f_i is an external forcing. The full state is $\mathbf{x} = \begin{pmatrix} x_1 & \cdots & x_N & \dot{x}_i & \cdots & \dot{x}_N \end{pmatrix}^T$, and we let $m = k = b = 1$. Thus, the system with full actuation and sensing is given by

$$\dot{\mathbf{x}} = \begin{pmatrix} \mathbf{0}_N & \mathbb{I}_N \\ \mathbf{T} & \mathbf{T} \end{pmatrix} \mathbf{x} + \begin{pmatrix} \mathbf{0}_N \\ \mathbb{I}_N \end{pmatrix} \mathbf{f} \tag{5.11a}$$

$$\mathbf{y} = \mathbb{I}_{2N}\mathbf{x}, \tag{5.11b}$$

where $\mathbf{0}_N$ is an $N \times N$ matrix of zeros and $\mathbf{T}$ is an $N \times N$ Toeplitz matrix with -2 along

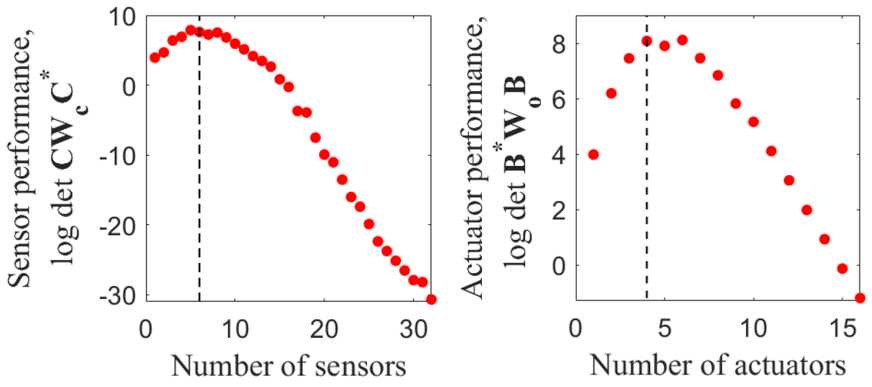

Figure 5.2: Sensor (actuator) performance metric as a function of the number of sensors (actuators) for the spring-mass system, with no cost function. The black dashed lines show the number of sensors and actuators we use in the example below.

the diagonal and 1 along the first super- and subdiagonals.

The goal is to place sensors by selecting rows of $\mathbf{C} = \mathbb{I}_{2N}$ and to place actuators by selecting columns of $\mathbf{B} = \begin{pmatrix} \mathbf{0}_N & \mathbb{I}_N \end{pmatrix}^T$, with cost functions on sensor and actuator locations.

We take $N = 16$ masses and place six sensors and four actuators. We find empirically that this leads to near-optimal results, as shown in Figure 5.2, which plots the sensor performance $\log\det \mathbf{C}\mathbf{W}_c\mathbf{C}^*$ as a function of the number of sensors on the left, and actuator performance $\log\det \mathbf{B}^*\mathbf{W}_o\mathbf{B}$ versus the number of actuators on the right, without incorporporating a cost function. The black dashed lines show the number of sensors and actuators we use. Although five sensors is optimal by a small margin, we prefer to select an even number of sensors, and therefore we choose six. Similarly, six actuators very slightly outperforms four, but we prefer to place fewer actuators than sensors, since there are fewer possible actuator locations.

We assume we have access to both position and velocity sensors and that the two types of sensors do not interfere with each other, so that the algorithm is free to select any six locations in $\mathbf{x}$. Since this is a toy example, we are free to select any reasonable cost function. We choose a Gaussian cost function for both position and velocity sensors, such that it is

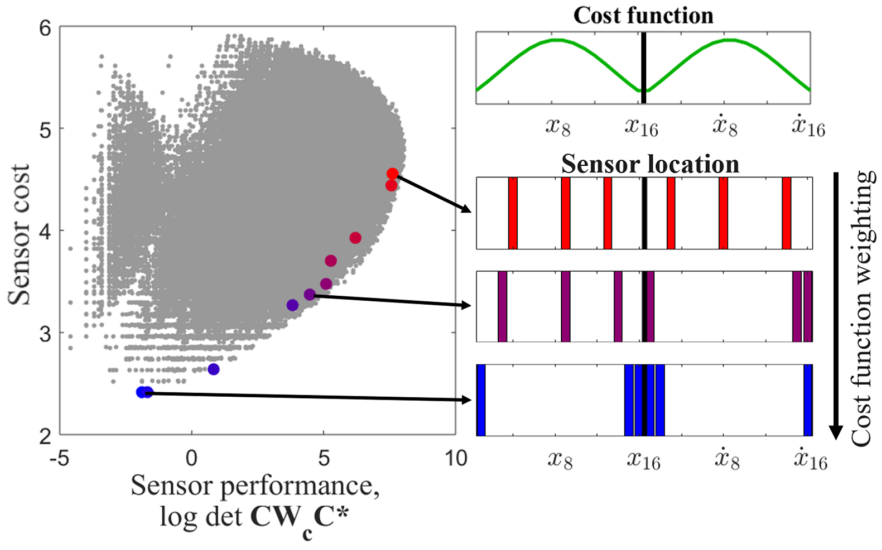

Figure 5.3: Sensor performance results and locations for the spring-mass system, placing 6 out of 32 possible sensors. The left plot shows cost versus performance metric log det $\mathbf{CW}_c\mathbf{C}^*$. The large colored data points show the modified-QR-selected sensors as the cost function weighting is increased, with red representing $\gamma = 0$ and blue representing $\gamma = 10$. The gray points give the cost and performance metric for every possible configuration of sensor selection. On the right are the sensor locations at three values of γ, where each bar corresponds to a sensor at the location x_i or $\dot{x}_i$, shown in the same colors as the indicated corresponding points on the left. The Gaussian cost function is shown on the top right. The black line indicates the separation between position and velocity variables in $\mathbf{x}$.

most expensive to place either type of sensor in the center of the array. We select an inverted Gaussian as the actuator cost function, making it more expensive to place actuators at the edges of the array. The cost functions are shown in the upper right-hand panels of Figure 5.3 and 5.4.

Because the actuation takes the form of an externally applied force $\mathbf{f}$, which causes a change in the velocity of the masses, the actuators are restricted to acting on the velocity coordinates, which can be seen in Equation 5.11a. Therefore, after calculating the truncated balanced and adjoint modes, we perform the cost-modified column-pivoted QR decomposition only on the final N rows of the adjoint mode matrix for actuator selection. Equivalently, we could set the first N rows to zero before performing the QR decomposition, or assign

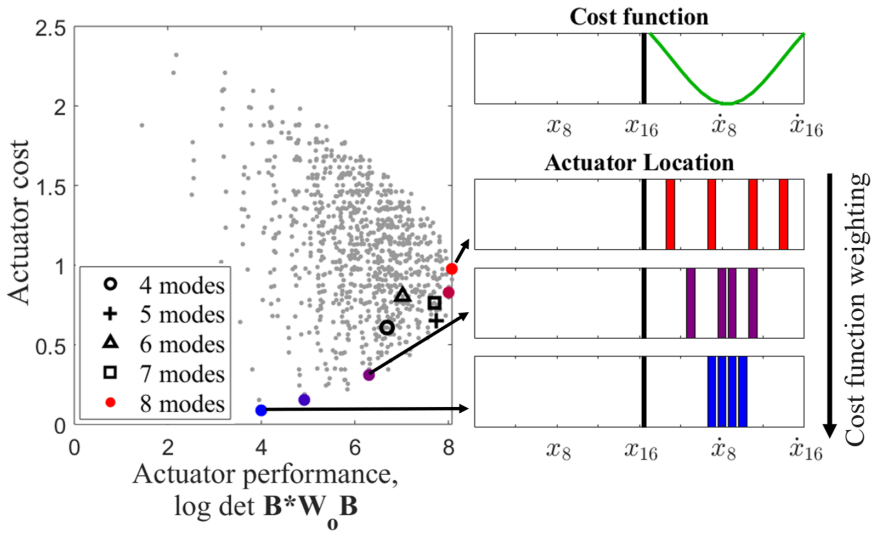

Figure 5.4: Actuator performance results and locations, placing 4 out of 16 possible actuators for the spring-mass system. Plotted on the left are cost versus performance metric log det $\mathbf{B}^*\mathbf{W}_o\mathbf{B}$. As in Figure 5.3, the large colored data points show the results with principled actuator selections, shaded from red to blue as the cost function weighting is increased from zero to ten, while the gray points give the cost and performance metric for every possible configuration of actuator selection. The black symbols show the cost and performance using four to seven balanced modes with $\gamma = 0$. The right plots show three sets of actuator locations, with the bars corresponding to actuator selections, as the cost function weighting is increased. Colors correspond to the indicated points on the left. The inverted Gaussian cost function is shown on the top right. Recall that actuators are restricted to the velocity variables, the locations to the right of the black bar on the plots.

those coordinates an infinite cost. Because this cuts the domain of possible actuators in half, we find that twice the number of modes are required to achieve good performance, i.e. to place four actuators, truncate at eight balanced adjoint modes before performing the QR decomposition.

Results in terms of the H_2-norm metric substitute are shown in Figure 5.3 and 5.4, which plot cost versus log det $\mathbf{C}\mathbf{W}_c\mathbf{C}^*$ and log det $\mathbf{B}^*\mathbf{W}_o\mathbf{B}$ for the sensor and actuator arrays, respectively. The large colored data points are the results from the arrays chosen by QR pivoting, with red corresponding to $\gamma = 0$, and blue indicating a large value of γ. The gray points show the results from every other combination of choosing 6 out of 32 sensors or 4

out of 16 actuators. The principled sensors approximately follow the curve of simultaneously maximal H_2 proxy and minimal cost. The best-performing configuration (cost function weighting set to zero) outperforms 99.8% of all other arrays, and with a high cost function weighting, the algorithm identifies the configuration with the lowest possible cost. Similarly, when the cost function weighting is set to zero, the selected actuators outperform all but three of the possible actuator selection permutations, and with a high value of γ, the algorithm again selects the lowest-cost permutation. Interestingly, the algorithm always selects three position and three velocity sensors.

The plots on the right-hand sides of Figure 5.3 and 5.4 show the sensor and actuator configurations for three different values of γ, in colors corresponding to those in the left panels. As the cost function weighting is increased, the sensors and actuators are pushed out of regions of high cost. The cost functions are also shown in green for reference. Finally, the left plot of Figure 5.4 also shows the performance of QR-selected actuators with $\gamma = 0$ and a number of balanced modes increasing from four to seven (shown in black, with symbols indicating the number of modes). Using eight modes yields the best results in terms of $\log\det \mathbf{B}^*\mathbf{W}_o\mathbf{B}$.

We also test sensor and actuator performance by evolving the system in time and applying linear-quadratic-Gaussian (LQG) control. To evaluate sensor performance, a small amount of system disturbance and noise are added in, both with covariance 0.005, and we construct a Kalman filter to obtain a state estimation $\hat{\mathbf{x}}(t)$. We integrate the system from time $t = 0$ to $t = 50$ and stack the true states and their estimations into the columns of data matrices as shown:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}(0) & \mathbf{x}(t_1) & \cdots & \mathbf{x}(50) \end{bmatrix} \tag{5.12a}$$

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{\mathbf{x}}(0) & \hat{\mathbf{x}}(t_1) & \cdots & \hat{\mathbf{x}}(50) \end{bmatrix}. \tag{5.12b}$$

The reconstruction error is then given by Equation 2.9b. Better sensor arrays will provide more informative measurements, and will therefore have lower reconstruction errors.

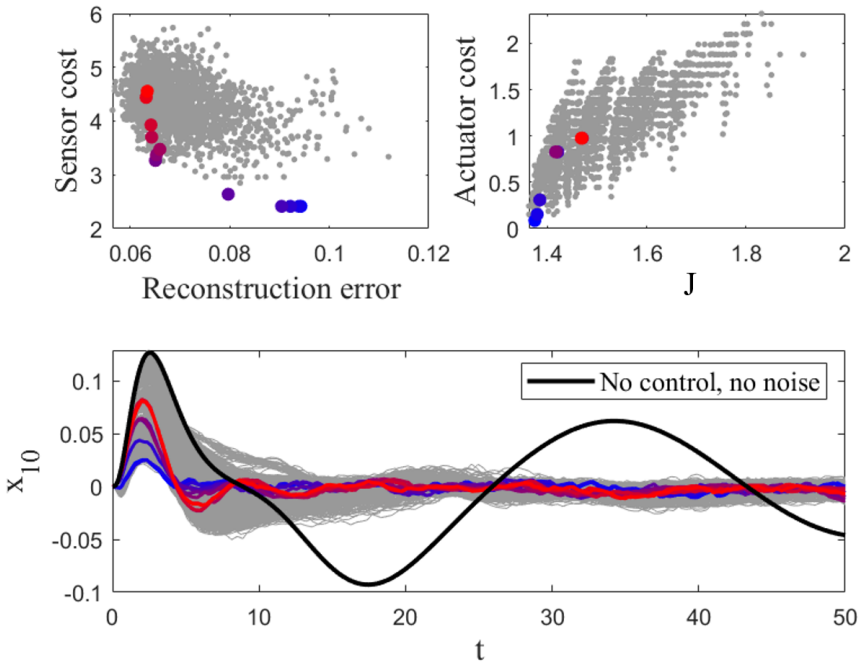

Figure 5.5: Sensor and actuator results using LQG control and state estimation for an example time series of the spring-mass system. The state is initialized at $\mathbf{x}(0) = 0$, except for $x_8(0) = 1$. The upper left plot shows sensor cost versus full-state reconstruction error. Red indicates $\gamma = 0$, blue corresponds to $\gamma = 10$, and gray shows results from 1820 random permutations of selecting 6 out of 32 possible sensors. The plot on the upper right similarly shows actuator cost versus the LQR actuation cost $J = \int_0^{50} \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right) dt$. The gray dots show all 1820 possible configurations of placing 4 out of 16 actuators. Finally, the trajectory $x_{10}(t)$ is plotted for the various sensor/actuator selections, with the black reference trajectory using no control and having no additive system or measurement noise.

At every time step, the state estimate is passed to a linear-quadratic regulator (LQR), designed to drive the entire system towards $\mathbf{x} = \mathbf{0}$. The performance of the actuator array is evaluated by calculating the LQR cost over the full timespan,

$$J = \int_0^{50} \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right) dt, \tag{5.13}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are identity matrices. Good actuator configurations will be able to bring the system to equilibrium quickly, and will have lower values of J.

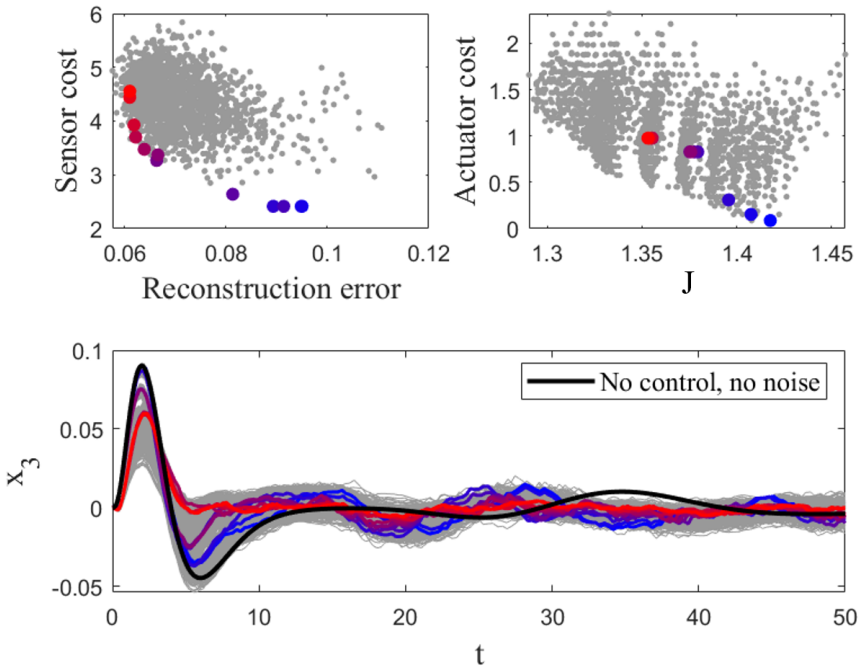

Figure 5.6: Sensor and actuator results for the spring-mass system using LQG control, with the inital condition $\mathbf{x}(0) = 0$, except for $x_1(0) = 1$. As in Figure 5.5, the upper left plot shows sensor cost versus reconstruction error, the upper right plot gives actuator cost versus $J = \int_0^{50} \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right) dt$, and the lower plot gives an example time series, in this case $x_3(t)$. Red points and lines indicate a cost function weighting of zero, blue indicates a high value of γ, and gray gives non-principled sensor and actuator selections. The black line in the lower plot shows $x_3(t)$ with no control or noise.

We test two different initial conditions, one with all positions and velocities set to zero except for the eighth mass, which has position $x_8(0) = 1$, and the other where $x_1(0) = 1$ with all other variables set to zero. Results are given in Figure 5.5 and 5.6. The figures have the same format, with the upper left subplot showing sensor cost versus reconstruction error, the upper right plot showing actuator price as a function of actuation cost J, and the lower subplot plotting the trajectory of one of the masses over time ($x_{10}(t)$ in Figure 5.5 and $x_3(t)$ in Figure 5.6). The colored dots and lines show the results with the principled sensor and actuator arrays, with the usual color scheme of red indicating $\gamma = 0$, shading through to blue for large values of γ. The gray points and lines are results from all 1820 permutations

for placing 4 out of 16 actuators, each using a random array of sensors (there are close to a million sensor placement permutations, so checking every possible sensor array with every possible actuator array is not feasible). The results for the upper plots are averaged over 25 realizations of noise, while the trajectories were all generated with the same noise vectors.

For both sets of initial conditions, the sensor configurations with $\gamma = 0$ provide effective, but not optimal, reconstructions compared to the randomly-selected arrays. As γ is increased, the cost decreases and the reconstruction error goes up by about 4% total. This can be seen qualitatively on the trajectory plots, where after $t \approx 20$, the red lines cluster more tightly about zero, while the blue lines have more variation.

As for the actuators, in the case of Figure 5.6, $x_1(0) = 1$, the QR-selected actuators are surprisingly far from optimal, though the performance decreases with the actuator cost, as expected. However, when $x_8(0) = 1$, Fig 5.5, the actuator array chosen with the largest value of γ leads not only to the lowest actuator cost but also to the optimum value of J. This seems counterintuitive, but because the center mass was given a large initial displacement, actuators placed in the center of the array are best suited to driving the entire system to zero in this case. Furthermore, the shape of the actuator cost versus J distributions is different than the sensor cost versus error distributions, and does not show a Pareto front. This is due to the difference in performance metrics. As described in Section 5.2, the QR algorithm acts as a greedy optimizer for the H_2 norm proxy, Equation 5.10. This metric is related to the reconstruction error, which is in turn closely related to the Kalman filter cost function, the expectation of the error covariance. However, the H_2 norm is not directly related to the LQR actuation cost function J, so we find that our actuator arrays are not optimal with respect to this performance metric. It may be that this method is suited only to sensor selection, and not actuator selection, though perhaps some modified balancing transformation that accounts for the directions of highest control cost as given by $\mathbf{Q}$ and $\mathbf{R}$ could improve results.

This example demonstrates that the column-pivoted QR decomposition on balanced modes for control systems can be modified to include a cost function on sensors and ac-

tuators. The method will select sensors and actuators to approximately simultaneously minimize cost and maximize performance, though for practical use the sensors lead to good reconstructions, but the actuators do not necessarily perform well in terms of the LQR cost function, depending on the initial conditions.

5.3 Sensor selection in a DMD basis

Dynamic mode decomposition (DMD) was introduced in the fluid dynamics community as a method of dimensionality reduction [87]. Conceptually, DMD is comparable to a spatial SVD combined with a temporal Fourier decomposition, in that it extracts spatially coherent modes that exhibit oscillatory or growth/decay behavior in time. This provides a highly effective reduced order model that can be used for short-time prediction, and can be adapted for applications such as control [101] and regime identification [102]. Furthermore, DMD has strong connections to the Koopman operator [88, 103]: DMD approximates the Koopman operator advancing the state forward in time. Systems are often sparse in a DMD basis [104], and DMD has been paired with compressed sensing in both space [105, 106] and time [107]. Sparse sensing for multiresolution DMD was formulated by Manohar et al. [91]. Here we provide a sparse sensing method for DMD modes allowing for a heterogeneous cost on sensor location.

We calculate the dynamic mode decomposition as follows [89]: We begin by collecting a set of m full-state snapshots, spaced equidistant in time (an alternative formulation does not require snapshots to be spaced equally in time [108]),

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}(t_1) & \mathbf{x}(t_2) & \cdots & \mathbf{x}(t_m) \end{bmatrix}. \tag{5.14}$$

We then organize the snapshots into two matrices offset from each other by one time step:

$$\mathbf{X}_1 = \begin{bmatrix} \mathbf{x}(t_1) & \mathbf{x}(t_2) & \cdots & \mathbf{x}(t_{m-1}) \end{bmatrix} \tag{5.15a}$$

$$\mathbf{X}_2 = \begin{bmatrix} \mathbf{x}(t_2) & \mathbf{x}(t_3) & \cdots & \mathbf{x}(t_m) \end{bmatrix}. \tag{5.15b}$$

The goal of DMD is to find the best-fit matrix $\mathbf{A}$ such that

$$\mathbf{X}_2 \approx \mathbf{A}\mathbf{X}_1, \tag{5.16}$$

i.e. the matrix $\mathbf{A}$ that advances the system forward by one time step. We take the r-mode truncated SVD of $\mathbf{X}_1$:

$$\mathbf{X}_1 = \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^*. \tag{5.17}$$

The full least squares solution of Equation 5.16 is given by

$$\mathbf{A} = \mathbf{X}_2\mathbf{X}_1^\dagger = \mathbf{X}_2\mathbf{V}_r\mathbf{\Sigma}_r^{-1}\mathbf{U}_r^*, \tag{5.18}$$

but since a primary goal of DMD is rank truncation, only the leading r eigenvalues and eigenvectors of $\mathbf{A}$ are of interest here. Therefore, we calculate the projection $\tilde{\mathbf{A}}$ of $\mathbf{A}$ onto the leading r spatial SVD modes,

$$\tilde{\mathbf{A}} = \mathbf{U}_r^*\mathbf{A}\mathbf{U}_r \tag{5.19a}$$

$$= \mathbf{U}_r^*\mathbf{X}_2\mathbf{V}_r\mathbf{\Sigma}_r^{-1}. \tag{5.19b}$$

We take the eigenvalue decomposition:

$$\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda}. \tag{5.20}$$

The eigenvalues along the diagonal of $\mathbf{\Lambda}$ are identical to the leading r eigenvalues of the full matrix $\mathbf{A}$, and the leading eigenvectors are given by

$$\mathbf{\Psi} = \mathbf{X}_2\mathbf{V}_r\mathbf{\Sigma}_r^{-1}\mathbf{W}. \tag{5.21}$$

The k^{th} state $\mathbf{x}_k = \mathbf{x}(t_k)$ can then be expressed in terms of these eigenmodes and eigenvalues:

$$\mathbf{x}_k = \mathbf{\Psi}\mathbf{\Lambda}^{k-1}\mathbf{b}, \tag{5.22}$$

or the continuous-time equivalent

$$\mathbf{x}(t) = \mathbf{\Psi}e^{\mathbf{\Omega}t}\mathbf{b}, \tag{5.23}$$

where $\mathbf{b}$ is the vector of amplitudes $\mathbf{b} = \mathbf{\Psi}^{\dagger}\mathbf{x}_1$, and $\mathbf{\Omega}$ is a diagonal matrix whose diagonal entries are the continuous-time eigenvalues $\omega_j = \log \lambda_j / dt$.

Thus, the eigenvectors $\mathbf{\Psi}$ are coherent spatial modes that form a reduced basis for the system of interest. We use this basis for sensor selection by performing the cost-modified column-pivoted QR decomposition on $\mathbf{\Psi}^*$.

5.3.1 Example

As an example system, take the NOAA sea surface temperature data set [70–72], as described in Section 3.3. The system is approximately periodic due to seasonal variations, so we expect DMD to give a reasonably good description of its behavior. To simulate a realistic sensor selection scenario, we again use a cost function that uniformly penalizes placing sensors more than two grid points from land. An example sea surface temperature snapshot and the cost function are shown in Figure 5.7.

We provide two performance metrics for the sensor arrays, interpolative error and extrapolative error:

$$E_{int} = \frac{||\mathbf{X}^{tr} - \hat{\mathbf{X}}^{tr}||_F}{||\mathbf{X}^{tr}||_F} \tag{5.24a}$$

$$E_{ext} = \frac{||\mathbf{X}^{te} - \hat{\mathbf{X}}^{te}||_F}{||\mathbf{X}^{te}||_F}, \tag{5.24b}$$

where the columns of $\mathbf{X}^{tr}$ are the training set, the first 80% of snapshots, which are used to

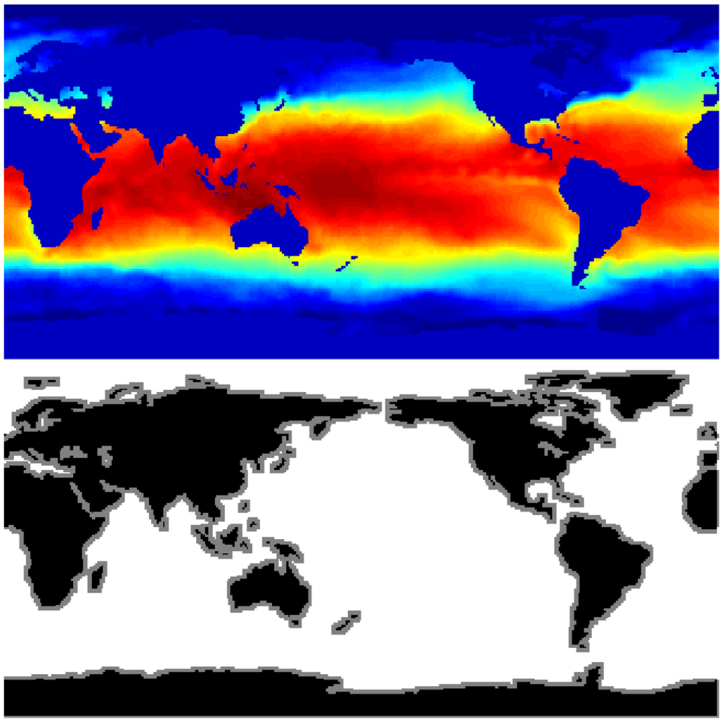

Figure 5.7: An example sea surface temperature snapshot (upper) and the cost function used (lower), where gray indicates regions of uniformly low cost and white denotes uniformly high cost.

train the DMD modes. The columns of $\hat{\mathbf{X}}^{tr}$ are their reconstructions. $\mathbf{X}^{te}$ and $\hat{\mathbf{X}}^{te}$ are the test set, the remaining 291 snapshots and reconstructions sequentially following the training set.

The sea surface temperature data set has a slow singular value decay with a Gavish-Donoho cutoff [76] around 400 modes. Thus we test our sensor selection method with both a small number of sensors (5) and a large number of sensors (200), though both cases represent significant downsampling from the original snapshots with dimension $\sim 10^5$. Our results are shown in Figure 5.8. Red data points have a cost function weighting of zero, while blue points have a large weighting, $\gamma = 1$. As the cost function is weighted more heavily, the sensor array cost decreases to zero, leading to a small increase in reconstruction error, though remarkably

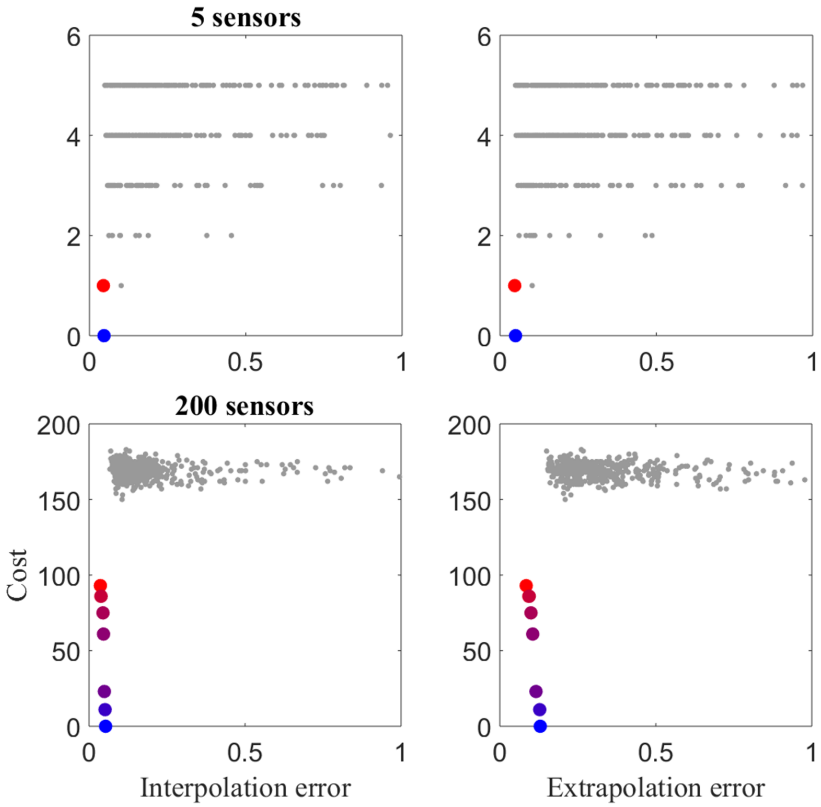

Figure 5.8: Cost versus reconstruction error for the sea surface temperature system using DMD modes as the cost function weighting is increased from zero (red) to one (blue). Results are shown for 5 sensors on the top row and for 200 sensors on the bottom row. The small gray points are results from 500 random sensor arrays.

this increase is by less than 1%. Results from 500 random sensor arrays are also shown for comparison in gray, since the system is far too large to check every possible permutation (there are $\sim 10^{20}$ ways to place just five sensors). The QR-based algorithm places most of the sensors close to land even without incorporating the cost function, so not only do the random sensors generally lead to higher reconstruction errors, they also have higher costs. As expected, interpolation errors are lower than extrapolation errors. Unexpectedly, there is little improvement in increasing from 5 to 200 sensors, and in fact extrapolation errors are higher with more sensors. This is most likely related to overfitting and noise amplification,

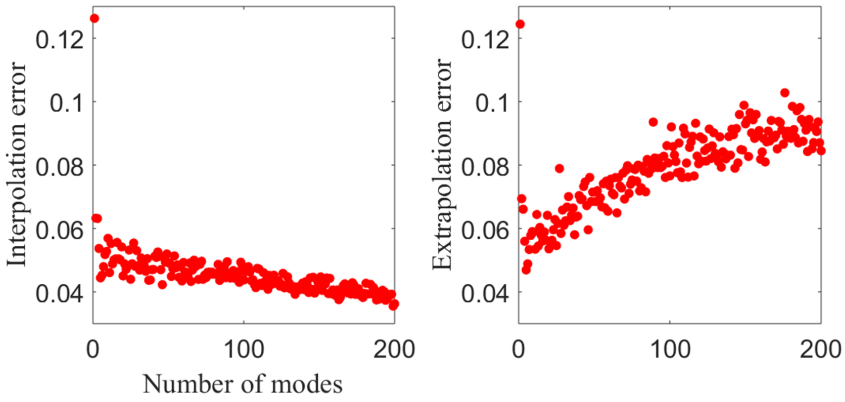

Figure 5.9: Error as a function of the number of DMD modes (equal to the number of sensors) for the sea surface temperature data set. Interpolation error is shown on the left and extrapolation error is on the right. No cost function is used.

see [77] for more details. Interpolation and extrapolation error with no cost function are plotted with r ranging from 1 to 200 in Figure 5.9. It is apparent that while interpolation error decreases slightly as the number of modes is increased, the extrapolation error increases significantly, as suggested by Figure 5.8.

We also provide a comparison to other basis choices in Figure 5.10. We test randomized rank reduction as described in [38], and take twice the number of modes as sensors for greatly improved performance. These results are plotted as open circles. Results with an SVD basis are shown as crosses, and we follow the guidelines of [77], performing the modified QR decomposition on the first $p/2$ SVD modes and placing the remaining $p/2$ sensors randomly to avoid noise amplification. Both of these basis choices lead to lower reconstruction errors at 200 sensors than the DMD basis, shown as closed circles, though the random sensors used with the SVD basis lead to higher costs. However, the benefit of DMD is the information it provides about the system's dynamics, which allows us to construct a Kalman estimator for the DMD amplitudes. From Equation 5.23, the time-varying amplitudes are given by

$$\mathbf{b}(t) = e^{\mathbf{\Omega} t}\mathbf{b}(0). \tag{5.25}$$

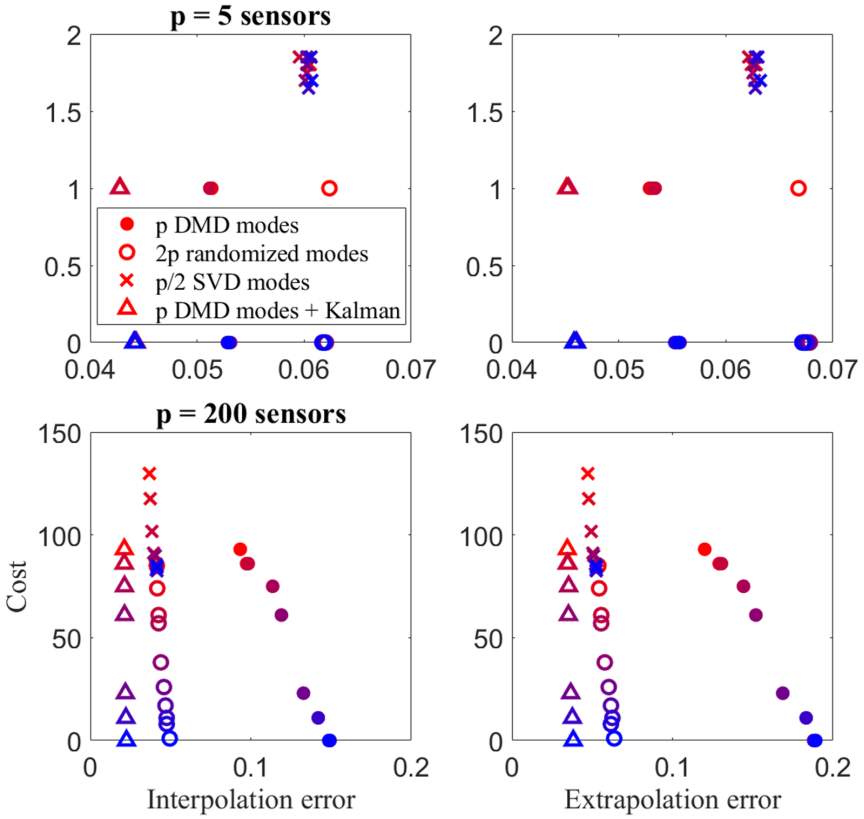

Figure 5.10: Cost versus reconstruction error for the sea surface temperature system, using DMD, randomized, and SVD modes with $p = 5$ and 200 sensors. For the DMD modes, we test both the usual linear algebra reconstruction method of Equation 2.7 and a Kalman estimator (triangles).

We can then evolve the system forward in time according to

$$\frac{d}{dt}\mathbf{b} = \mathbf{\Omega}\mathbf{b}. \tag{5.26}$$

We include added sensor noise with 2% variance (this noise is applied to all of the sensor arrays for Figure 5.10). The Kalman estimator, results plotted as triangles, significantly outperforms the other three methods, suggesting that sensors and reconstruction methods that account for the dynamics of a system are preferable when possible.

5.4 Sensor selection in an analytic basis

In many physical systems, a tailored data-driven basis is not necessary because the system is well-described by a universal basis, or because its equations of motion are known and can be solved analytically to yield a modal decomposition. It is then possible to write the state according to Equation 2.2, where $\mathbf{a}$ are Fourier coefficients and $\mathbf{\Psi}$ is a DFT matrix with $r = n$. Then the sparse sensing problem is similar to compressed sensing [20], in which a universal basis with random measurements is used to reconstruct a signal. Here, however, the signal is not required to be sparse in $\mathbf{\Psi}$, since we use least squares, rather than L_1-regularized reconstruction. Other examples of Equation 2.2 include polynomial interpolation and Gaussian quadrature.

Alternatively, the system can be time-varying:

$$\mathbf{x}(t) = \mathbf{\Psi}\mathbf{a}(t). \tag{5.27}$$

This is mathematically similar to the DMD formulation of Equation 5.23, the main differences being that the spatial modes are here assumed to be found analytically, rather than through data-driven methods, and we make no assumptions about the form of the time dependence.

As in previous sections, once $\mathbf{\Psi}$ is calculated, Algorithm 2 can be applied to $\mathbf{\Psi}^*$ for cost-constrained sensor selection.

5.4.1 Example

We test the performance of an analytic basis for sensor selection on a vibrating two-dimensional circular membrane, since the equations of motion for this system are known and can be solved for a modal decomposition. The membrane satisfies the following partial differential equation:

$$\frac{\partial^2 u}{\partial t^2} = c^2 \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial u}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2 u}{\partial \theta^2} \right), \tag{5.28}$$

with $\theta \in (-\pi, \pi]$, $r \in [0, a]$. We take the boundary and initial conditions

$$u(a, \theta, t) = 0 \tag{5.29a}$$

$$u(r, \theta, 0) = \alpha(r, \theta) \tag{5.29b}$$

$$\frac{\partial u}{\partial t}(r, \theta, 0) = 0. \tag{5.29c}$$

This system can be solved analytically through separation of variables. The solution is a Bessel series in r and a Fourier series in θ and t:

$$u(r, \theta, t) = \sum_{n=1}^{\infty} A_{0n} J_0\left(\sqrt{\lambda_{0n}} r\right) \cos\left(c\sqrt{\lambda_{0n}} t\right) + \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \left(A_{mn} \cos m\theta + B_{mn} \sin m\theta\right) J_m\left(\sqrt{\lambda_{mn}} r\right) \cos\left(c\sqrt{\lambda_{mn}} t\right), \tag{5.30}$$

where $\lambda_{mn} = (z_{mn}/a)^2$, with z_{mn} the n^{th} zero of the m^{th} Bessel function.

Thus, the basis modes of the system are $\cos(m\theta) J_m\left(\sqrt{\lambda_{mn}} r\right)$ and $\sin(m\theta) J_m\left(\sqrt{\lambda_{mn}} r\right)$, forming the columns of the basis. Practically, the basis matrix must be finite, so we truncate at $m = M$, $n = N$ modes:

$$\begin{aligned} \boldsymbol{\Psi}_{MN} = \Big(& J_0\left(\sqrt{\lambda_{01}} r\right) \quad \cdots \quad J_0\left(\sqrt{\lambda_{0N}} r\right) \quad \cos(\theta) J_1\left(\sqrt{\lambda_{11}} r\right) \quad \sin(\theta) J_1\left(\sqrt{\lambda_{11}} r\right) \quad \cdots \quad \cos(\theta) J_1\left(\sqrt{\lambda_{1N}} r\right) \\ & \sin(\theta) J_1\left(\sqrt{\lambda_{1N}} r\right) \quad \cdots \quad \sin(M\theta) J_M\left(\sqrt{\lambda_{MN}} r\right) \Big). \end{aligned} \tag{5.31}$$

The entire system evolves as

$$\mathbf{u}(t) = \boldsymbol{\Psi}_{MN} \cos\left(c\sqrt{\boldsymbol{\Lambda}_{MN}} t\right) \mathbf{b}_{MN}, \tag{5.32}$$

where $\boldsymbol{\Lambda}_{MN}$ is a diagonal matrix with entries $\lambda_{01}, \ldots, \lambda_{0N}, \lambda_{11}, \lambda_{11}, \lambda_{12}, \lambda_{12}, \ldots, \lambda_{1N}, \lambda_{1N}, \ldots, \lambda_{MN}, \lambda_{MN}$, and the cosine is taken elementwise on the diagonal entries. The vector $\mathbf{b}_{MN}$ gives the coef-

ficients:

$$\mathbf{b}_{MN} \quad = \quad \left(\; A_{01} \;\; \cdots \;\; A_{0N} \;\; A_{11} \;\; B_{11} \;\; \cdots \;\; A_{1N} \;\; B_{1N} \;\; \cdots \;\; A_{MN} \;\; B_{MN} \; \right)^T. \quad (5.33)$$

Our goal is to sparsely sample in space and then estimate the time-varying coefficients $\cos\left(c\sqrt{\mathbf{\Lambda}_{MN}}t\right)\mathbf{b}_{MN}$, subsequently obtaining a reconstruction of the full state $\hat{\mathbf{u}}(t)$. We select sensors by performing the cost-constrained column-pivoted QR decomposition on $\mathbf{\Psi}_{MN}^T$.

For this experiment, we take $a = 10$, $c = 1$, and a grid of 101 points each in r and θ. We also truncate the modes at $M = 6$ and $N = 5$ for a total of 55 basis modes. The system is initialized with random coefficients, with lower modes weighted more heavily, i.e.

$$A_{mn}, B_{mn} \propto \frac{1.5}{n(m+1)}. \quad (5.34)$$

Not only does this lead to more accurate reconstructions when undersampling, it is also often the case in physical systems that the lower modes are the most active. We choose a radially symmetric cost function with a maximum in the center and a minimum at $r = 13/2$, $f(r) = 0.6 + 0.5\cos\left(\frac{2\pi r}{13}\right)$, pictured in the lower right panel of Figure 5.11.

We take $\gamma \in [0, 20]$, and evaluate sensor performance by propagating the initial conditions forward in time steps of 0.1 up to $t = 10$. The true states are stacked into the columns of a snapshot matrix $\mathbf{U}$, and the error is given by $||\mathbf{U}-\hat{\mathbf{U}}||_F/||\mathbf{U}||_F$, where $\hat{\mathbf{U}}$ is the reconstruction obtained with the chosen sensor array. We calculate this error with 50 different sets of random initial conditions and take the average, plotted in the first column of Figure 5.11, which shows sensor cost plotted against this average error as γ is increased (red data points correspond to $\gamma = 0$, blue show $\gamma = 20$). The remaining panels in the figure show reconstructions of an example state with the sensor arrays chosen with various numbers of sensors and values of γ. The true state is shown on the bottom left of the figure. Even with just 10 sensors, when $\gamma = 0$, the reconstruction is imperfect but retains some main features of the true state. With 55 sensors, the state can be reconstructed exactly. As the cost function weighting increases,

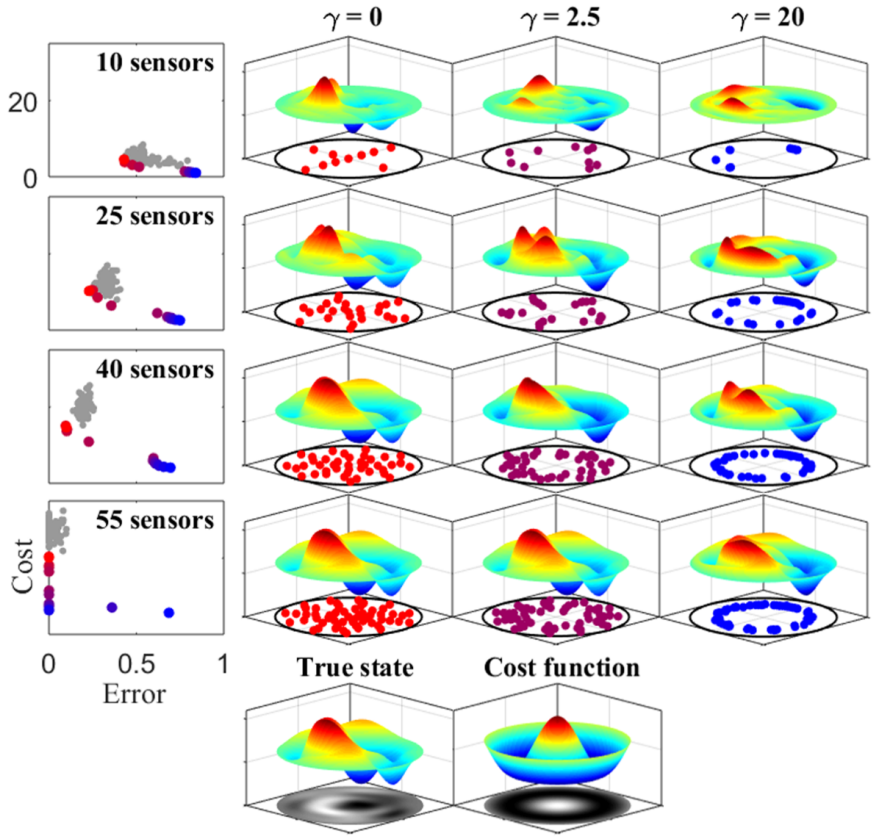

Figure 5.11: Results for sensor selection with a cost function for the vibrating drum system with 55 Bessel function basis modes. The number of sensors increases moving down the rows. The first column shows the cost versus reconstruction error as the cost function weighting is increased (red points indicate $\gamma = 0$, blue points have high values of γ, gray points show 100 random sensor arrays). Note that all plots in the first column have the same x- and y-axis scales. The remaining three columns show reconstructions of an example snapshot with the indicated number of sensors and value of γ. The sensor locations are given as well, with color corresponding to that on the cost versus error plots. The true state and the cost function are shown on the bottom row, in both a surface and a pcolor plot.

the reconstruction quality degrades, even with 55 sensors. A large jump in reconstruction error occurs when there are no longer any sensors close to the center of the domain. The figure also shows results from 100 random sensor arrays, which suggest that the principled sensors follow the curve of minimum cost and reconstruction error quite well, though the system is too large to check every possible sensor permutation.

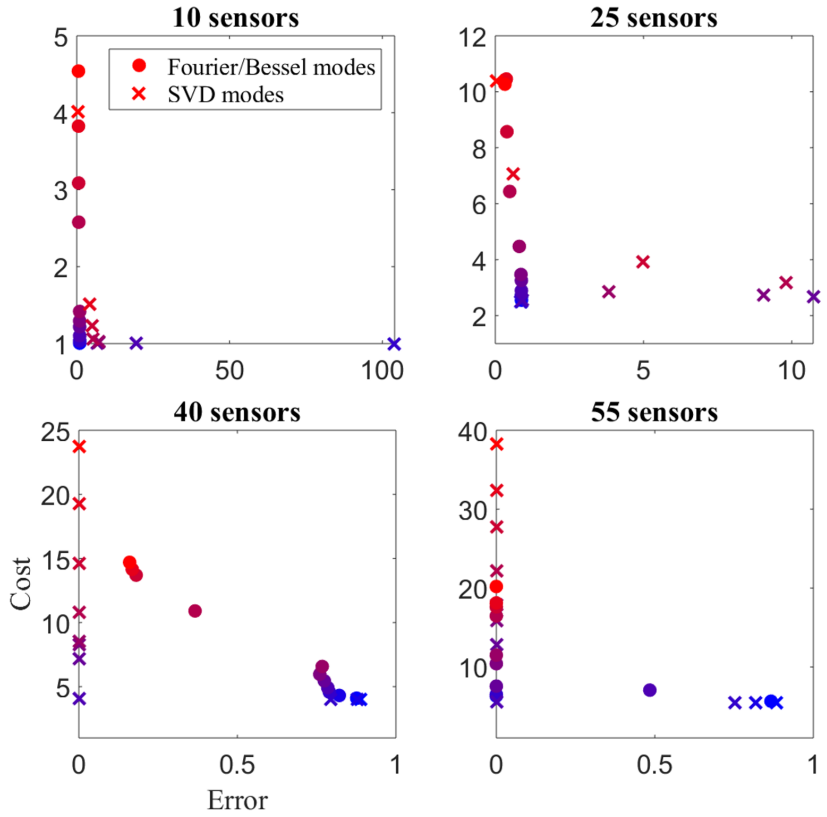

Figure 5.12: Cost versus reconstruction error for the vibrating drum system, using the analytic Fourier/Bessel modes of Equation 5.31 and SVD modes calculated from snapshots. As in Figure 5.11, results are shown for different numbers of sensors and multiple cost function weightings, with the usual color scheme of red indicating $\gamma = 0$ shading through to blue representing a large value of γ.

We also compare reconstruction results with sensors selected using an SVD basis, shown in Figure 5.12, which plots cost versus reconstruction error for 10, 25, 40, and 55 sensors. Sensors selected with analytic Fourier/Bessel modes are shown as circles, with SVD results shown as crosses. In both cases, red indicates a cost function weighting of zero and blue corresponds to a large cost function weighting. To calculate the SVD basis, we initialize a random system with coefficients as in Equation 5.34 and evolve it forward for 1000 time steps to obtain snapshots. We then randomly place 700 snapshots into a training matrix and

perform the SVD to obtain a basis. After using this basis and Algorithm 2 to select sensors, we test them on the remaining 300 snapshots, and compare to reconstructions of the same snapshots using sensors from the analytic basis. We take the same number of SVD modes as sensors since there is no noise in the system, and take the average cost and error over 50 random training and test sets.

As demonstrated in Figure 5.12, while the SVD and analytic modes have comparable performance at a small number of sensors, the SVD basis is able to reconstruct the system perfectly to machine precision at 40 sensors. With the SVD basis, the active modes are mixed together, which is more beneficial for rank reduction—with sensors selected using analytic modes, even with 54 sensors there may be little information about the 55^{th} mode. This can be seen clearly in Figure 5.13, which shows error versus the number of sensors using both bases and no cost function. The SVD basis consistently outperforms the analytic basis, requiring just 30 sensors to reconstruct the full state perfectly, whereas the analytic modes need the full 55. However, the SVD requires data to calculate the basis, so in cases where high-fidelity full state data is not available, we have shown that analytically calculated modes are a viable alternative. Furthermore, the figure shows comparisons to random sensors using both bases for reconstruction. In the analytic basis, principled sensors have lower reconstruction errors than random sensors, and random sensors perform very poorly in an SVD basis, highlighting the importance of choosing the right method for the application at hand.

The results in this work indicate that if a dynamical system may be represented as the evolution of a superposition of known basis modes, these modes can be used for sparse sensor selection, even with a cost function on sensor location. Though a data-driven SVD basis leads to better sensor arrays, analytic modes often give comparable reconstructions. In a more complex signal, such as a sound signal being represented in a Fourier basis, where both high and low frequencies can be equally present and it is unknown which frequencies will be dominant, it is difficult to effectively downsample using this method. The problem is similar to that of compressed sensing, where in most real-world examples, the number of sensors required to reconstruct the signal with high probability is large and leads to a

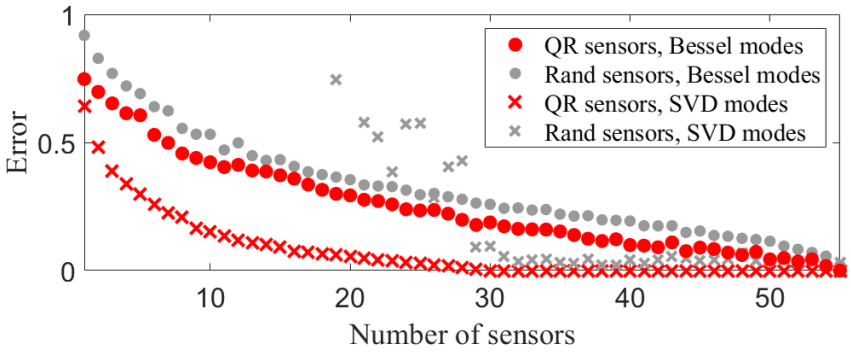

Figure 5.13: Error versus the number of sensors for the vibrating drum system, using both analytic Fourier/Bessel modes and data-driven SVD modes. Principled sensors selected by the QR algorithm are shown in red, compared to random sensors with both bases in gray. No cost function is used.

computationally prohibitive reconstruction problem [84]. These examples demonstrate how data-driven methods are usually more efficient at extracting patterns from a system than historical analytic-function-based methods.

5.5 Conclusions

In this chapter, we explore cost-constrained QR pivoting for sensor selection, applying the algorithm to three new types of basis modes: balanced modes for sensor and actuator selection for control systems, DMD modes for dynamical systems, and universal or analytic modes. We find that all three types of modes make effective bases for sensor selection for their respective systems, and that the cost-constrained algorithm selects sensors that are nearly Pareto optimal in cost and performance. The new bases do not necessarily outperform a standard SVD basis, but they can still be beneficial. Sensors placed using a DMD basis can be used to construct a Kalman estimator, which does perform better than the SVD or randomized bases when measurement noise is present. And when no full-state data is available, analytic modes can be a good substitute for a data-driven basis.

These results suggest that sensor selection methods must account for real-world restric-

tions. A researcher or engineer can improve their full state estimations by selecting an appropriate basis, but they must also consider practicalities like a sensor cost landscape, the purpose of the sensors (acutation versus reconstruction), the dynamics of the system being studied, and the availability of full-state training data.

Chapter 6

MULTI-FIDELITY SENSOR SELECTION: GREEDY ALGORITHMS TO PLACE CHEAP AND EXPENSIVE SENSORS WITH COST CONSTRAINTS

We develop greedy algorithms to approximate the optimal solution to the multi-fidelity sensor selection problem, which is a cost constrained optimization problem prescribing the placement and number of cheap (low signal-to-noise) and expensive (high signal-to-noise) sensors in an environment or state space. Specifically, we evaluate the composition of cheap and expensive sensors, along with their placement, required to achieve accurate reconstruction of a high-dimensional state. We use the column-pivoted QR decomposition to obtain preliminary sensor positions. How many of each type of sensor to use is highly dependent upon the sensor noise levels, sensor costs, overall cost budget, and the singular value spectrum of the data measured. Such nuances allow us to provide sensor selection recommendations based on computational results for asymptotic regions of parameter space. Our extensive exploration of multi-fidelity sensor composition as a function of data characteristics is the first of its kind to provide guidelines towards optimal multi-fidelity sensor selection.

6.1 Introduction

One facet of the sensor selection problem rarely addressed is the optimal placement of multiple types of sensors that can have significantly different costs and performance metrics. Sensors often measure different quantities (multi-modal sensing), or they could measure the same quantity but with different qualities (multi-fidelity sensing), i.e. different noise level, bias, sampling rate, range, power consumption, construction cost, placement cost, or longevity. Given access to two or more types of sensor, what principled mathematical method

can determine how many of each type to select? Where should they be placed in order to optimize relevant factors such as performance and cost? This chapter considers the sparse selection of multi-fidelity sensors with each having significantly different costs and signal-to-noise levels. Specifically, our proposed sparse selection procedure is aimed at determining the optimal number of each sensor type and their placement so as to simultaneously minimize reconstruction error and financial cost.

Because of the wide range of sparse sensing applications, there is a significant body of theoretical work on principled sensor selection spanning many disciplines, see Chapter 1 for some prominent examples. However, there is typically a disconnect between mathematical methods and practical implementations of sensor selection. Specifically, it is usual to assume one type of sensor, idealized conditions, and the availability of full-state training data, while neglecting important factors like sensor type, noise, and cost. Bridging the gap between idealized models and real-world practicalities is currently an open problem.

Importantly, multi-fidelity sensor placement has received little consideration, and this is a central aim of our study. The goal for multi-fidelity sensor placement is to judiciously use expensive, high-performance sensors in combination with cheap, low-performance sensors to provide accurate state-space reconstructions as cheaply as possible. For us, the difference between cheap and expensive sensors will be associated with the additive noise at measurement. Cheap sensors have large additive noise, and expensive sensors have low additive noise.

The literature on the optimal selection of multiple types of sensors is somewhat limited. Lahat et al [109] advocate for the benefits of multimodality for state estimation, and the structure monitoring community makes use of multiple types of sensors for fault detection and response reconstruction. Sparse strain gauges, accelerometers, and displacement transducers are selected by means of greedy [110–112] or genetic [113, 114] algorithms. In the case of sensors that measure the same quantity with different qualities, Kammer [115] describes a method for sensor selection in the case of location-dependent noise levels. For energy-harvesting Internet of Things networks, convex optimization formulations have been

developed that account for sensor noise level, transmission power, and energy harvesting capabilities [116–118]. These methods are powerful, but a convex relaxation is just one possibility for handling the overarching NP-hard problem of sensor selection. In this chapter, we continue to use the column-pivoted QR decomposition to obtain sensor locations, which is a fast, greedy algorithm that is available on any standard software package.

The multi-fidelity optimization procedure can be stated colloquially as follows: Is it better to purchase a large number of cheap sensors, a small number of expensive sensors, or a mix of both? In the latter case, how many of each type should be used, and where should they be placed? Alternatively, someone may already have sensors with low signal-to-noise levels, and needs to determine whether it is worth the time and money to design sensors with higher signal-to-noise levels. Quantitative answers to these questions can not only address feasibility issues, but can potentially save manufacturers significant amounts of money. However, the multi-fidelity sparse sensor selection problem is complicated and nuanced, depending on the rank of the data considered, the relative and absolute costs and noise levels of the sensors, and the available budget.

In this work, we characterize optimal sensor selection with two types of sensors in a few asymptotic regimes of very low or very high costs and noise levels, for systems whose data is of low, medium, and high rank, given a fixed budget. We present empirical results and use them to formulate an initial set of guidelines for selecting a sensor type. Broadly, if both types of sensors have low noise levels, it is usually better to use a larger number of cheap sensors. When the cheap sensors have a much higher noise level than the expensive sensors, a small number of expensive sensors tends to perform better. And when both sensors have similar, high noise levels, the rank of the system is the determining factor, with high-rank systems favoring a larger number of cheap sensors, and low-rank systems preferring a smaller number of expensive sensors. Thus to evaluate the multi-fidelity sparse sensor selection problem, the nature of the measurement data plays a significant role. To our knowledge, this is the first systematic study of the role of multi-fidelity, cost constrained sensor selection as a function of the data, cost, and noise. We intend this study to serve as a starting point for the discovery

of a coherent rule set for multi-fidelity sensor selection, which should also include expansions such as the selection of more than two types of sensor.

Section 6.2 below describes the multi-fidelity sensor selection problem and presents results and general guidelines from several asymptotic regimes. The discussion and conclusions are given in Section 6.4.

6.2 Multi-fidelity sensor selection

The basic sensor selection problem setup and the column-pivoted QR decomposition solution we employ are the same as in Chapter 2, but we now consider the central aim of this chapter: multi-fidelity sensor selection. Specifically, cheap and expensive sensors will simply be characterized by the additive noise in a measurement. Cheap sensors will have higher additive noise than expensive sensors. Thus the cost of an expensive sensor improves the signal to noise ratio for the measurement.

We now allow for the presence of inhomogeneous measurement noise:

$$\mathbf{Y}(J) = \mathbf{C}_J \mathbf{X}^{te} + \boldsymbol{\epsilon}, \tag{6.1}$$

where the elements of $\boldsymbol{\epsilon} \in \mathbb{R}^{p \times \ell}$ are given by $\epsilon_{ji} \propto \mathcal{N}(0, \sigma_j^2)$. Once the noisy measurements have been obtained, the full state reconstruction is identical to Equation 2.7. For the purposes of this chapter, let the sensor noise variances take one of two values, $\sigma_j \in \{\sigma^{(exp)}, \sigma^{(ch)}\}$, with $\sigma^{(exp)}$ corresponding to expensive sensors and $\sigma^{(ch)}$ to cheap sensors. This implies $\sigma^{(exp)} < \sigma^{(ch)}$.

We choose a basis of SVD modes for $\boldsymbol{\Psi}$, which, as demonstrated in Chapter 4, provides nearly identical reconstruction errors to randomized rank reduction but with far fewer modes. We use $r = p/2$ modes (rounded up when p is odd) and randomized oversampling, which is orders of magnitude faster than principled oversampling and performs only slightly less well. Therefore, to place a total of p sensors, we perform the column-pivoted QR decomposition on the first $p/2$ modes and use the first $p/2$ pivots, then randomly place the remaining

$p/2$ sensors. Now we must further consider the distributions of the $p^{(ch)}$ cheap and $p^{(exp)}$ expensive sensors given the selected locations (where $p^{(exp)} + p^{(ch)} = p$). For example, the expensive sensors could be placed at the first $p^{(exp)}$ locations:

$$\sigma_{1:p^{(exp)}} = \sigma^{(exp)}, \quad \sigma_{p^{(exp)}+1:p} = \sigma^{(ch)}, \tag{6.2}$$

or the expensive sensors could be placed on the last set of selected locations:

$$\sigma_{1:p^{(ch)}} = \sigma^{(ch)}, \quad \sigma_{p^{(ch)}+1:p} = \sigma^{(exp)}. \tag{6.3}$$

There is logic behind either choice. Since the QR algorithm selects pivots in order of approximate importance, measuring the first set of selected locations with high accuracy should lead to better overall reconstructions. However, locations are chosen to have iteratively maximal variance, so that the final set of locations, although randomly chosen in this case, almost certainly has a smaller variance than the first set. Thus the first set of QR selected locations has a larger signal-to-noise ratio for a given sensor, meaning that even cheap noisy sensors should be able to capture these modes relatively accurately, while using expensive sensors to accurately measure the last set of locations means better resolving fine details on the reconstruction.

In fact, the expensive sensors can be placed in any configuration, and we find that the optimal distribution depends on basis choice and the parameters of the particular problem. However, for the examples and conditions considered below, the performance is slightly better when the expensive sensors are placed on the first set of locations, following Eq. 6.2. In the asymptotic cases we will consider below, it will be better to use just one type of sensor, so the choice is less important. Therefore, for clarity, we only show results from placing expensive sensors on the first set of sensor locations.

The parameter space for the multi-fidelity sensor selection problem also includes the cheap and expensive sensor noise levels $\sigma^{(ch)}$ and $\sigma^{(exp)}$, and the costs of both types of sensor, $c^{(ch)}$

and $c^{(exp)}$. We assume there is a set budget B, such that

$$c^{(ch)}p^{(ch)} + c^{(exp)}p^{(exp)} \leq B. \tag{6.4}$$

Depending on all of these factors, it may be best to use all cheap sensors, all expensive sensors, or a mixture of both. We will explore different parameter regimes and their effect on what type of sensor to use and where to place them in the next section.

6.3 Multi-fidelity sensor results

Because the parameter space is combinatorially large, we consider only a few asymptotic regimes and determine in these cases whether all cheap or all expensive sensors perform better. The cases explored below fall into one of nine categories comprising combinations of very low and very high $\sigma^{(exp)}$, $\sigma^{(ch)}$, $p^{(exp)}$, and $p^{(ch)}$.

We also wish to account for different kinds of data, so we construct three data sets with slow, medium, and fast singular value decays. We do this by using the left and right singular vectors from the Yale B Face data set combined with artificial singular values. We fit a power law function, $y = ax^b$, to the true singular values of the eigenface data set and find that they have the approximate form

$$y = (1.21 \times 10^5)x^{-1.14}. \tag{6.5}$$

We assume that the exponent $b = -1.14$ corresponds to medium singular value decay, and construct the artificial data sets with $b = -0.6, -1.1$, and -1.6 (retaining $a = 1.21 \times 10^5$). The singular values are plotted in Figure 6.1. The data set with $b = -1.6$ has low rank, with 90% of the system's energy being captured by just 23 modes. The data set with $b = -1.1$ needs 355 modes, while the data set with $b = -0.6$ is high rank, requiring 798 modes to capture 90% of its energy.

For reference, reconstruction results with one type of sensor are shown for the three

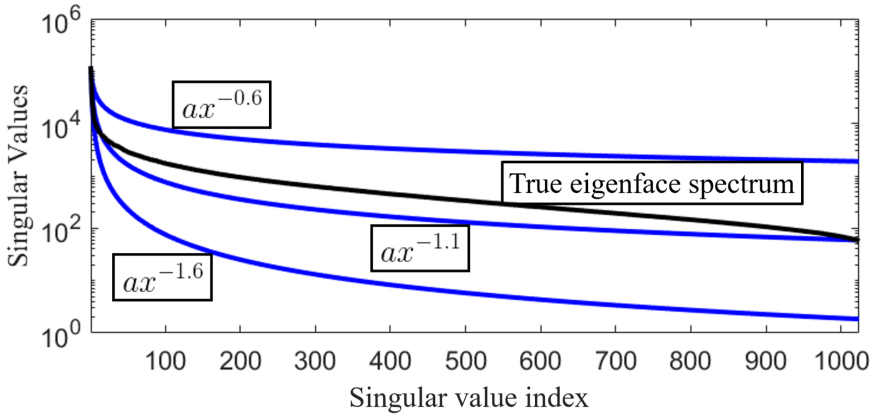

Figure 6.1: The singular values of the eigenface data set, plotted in black, and the artificial singular value spectra used in this section, shown in blue.

artificial data sets in Figure 6.2. Each subplot shows reconstruction error plotted against the number of sensors for one of the three data sets. We consider two high and two low noise levels. When $\sigma = 1\%$ and 2%, the data sets with fast and medium decaying singular values have very low reconstruction errors. For both high noise level cases, these data sets have very high errors and exhibit noise amplification, due to a small signal-to-noise ratio combined with a condition number greater than one [25]. The $b = -0.6$ data set always has high reconstruction errors, but the error always decreases as more sensors are added.

The main results of this paper are shown in Figure 6.3. The figure is split into nine sections, where the left column has low noise ($\sigma^{(exp)} = 1\%$, $\sigma^{(ch)} = 2\%$), the middle column has low $\sigma^{(exp)} = 1\%$ and high $\sigma^{(ch)} = 40\%$, and the right column has high-noise sensors, with $\sigma^{(exp)} = 35\%$ and $\sigma^{(ch)} = 40\%$. The top row has a small number of maximum allowed sensors, 2 and 4 for expensive and cheap, respectively. The middle row uses a maximum of 2 expensive and 400 cheap sensors, and the bottom row has a maximum of 300 expensive and 400 cheap sensors. In each section, results from the three data sets are plotted separately. From left to right they have $b = -1.6$, -1.1, and -0.6. Each of the panels shows reconstruction error as the proportion of cheap and expensive sensors is varied. Along the x-axis, the number of expensive sensors increases from zero to $p^{(exp,max)}$ (marked "E"), while the number of cheap

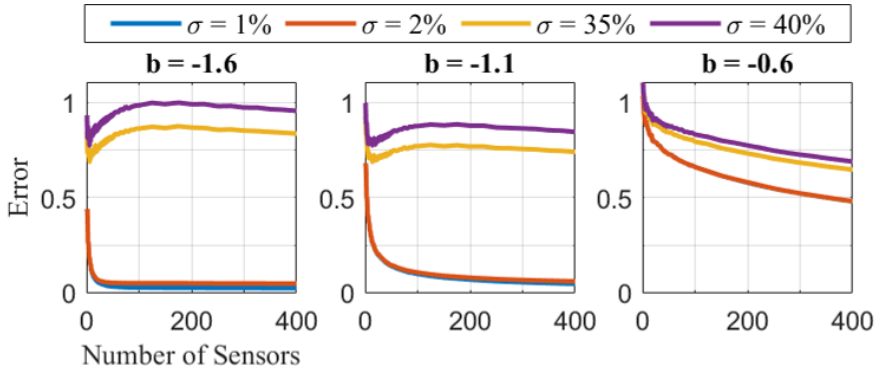

Figure 6.2: Reconstruction error versus number of sensors for three data sets with three different singular value exponents b. Results are given with just one type of sensor at a time, with four different noise levels, two very low and two very high.

sensors decreases from $p^{(ch,max)}$ (marked "C") to zero. At the midpoint, a mix of cheap and expensive sensors is used. All combinations satisfy Equation 6.4, where the budget and sensor costs are adjusted to satisfy both set values of $p^{(exp,max)}$ and $p^{(ch,max)}$ for every row. Results are averaged over 20 randomized training and test sets, each with 20 cross validations over sensor location and 10 noise realizations.

If the error is lower at point "C", then it is better to use all cheap sensors and the plot is colored red. Blue plots have lower reconstruction errors at "E", meaning that it is preferable to use all expensive sensors. The plots are shaded from dark to light, with dark indicating that there is a large difference in reconstruction error between the two types of sensors, and light meaning that the reconstruction errors are similar. Finally, plots are colored white when there is less than a 2% difference in using all cheap and all expensive sensors.

Although we have not yet found a rule for when to use all cheap versus all expensive sensors, the figure does reveal some trends. When both noise levels are very low, it is almost always better to use a large number of cheap sensors. When $\sigma^{(exp)}$ is low and $\sigma^{(ch)}$ is high, or when both sensor types have high noise levels, it is more often better to choose a small number of expensive sensors.

There are exceptions to the general trends, and it is clear that the rank of the data set

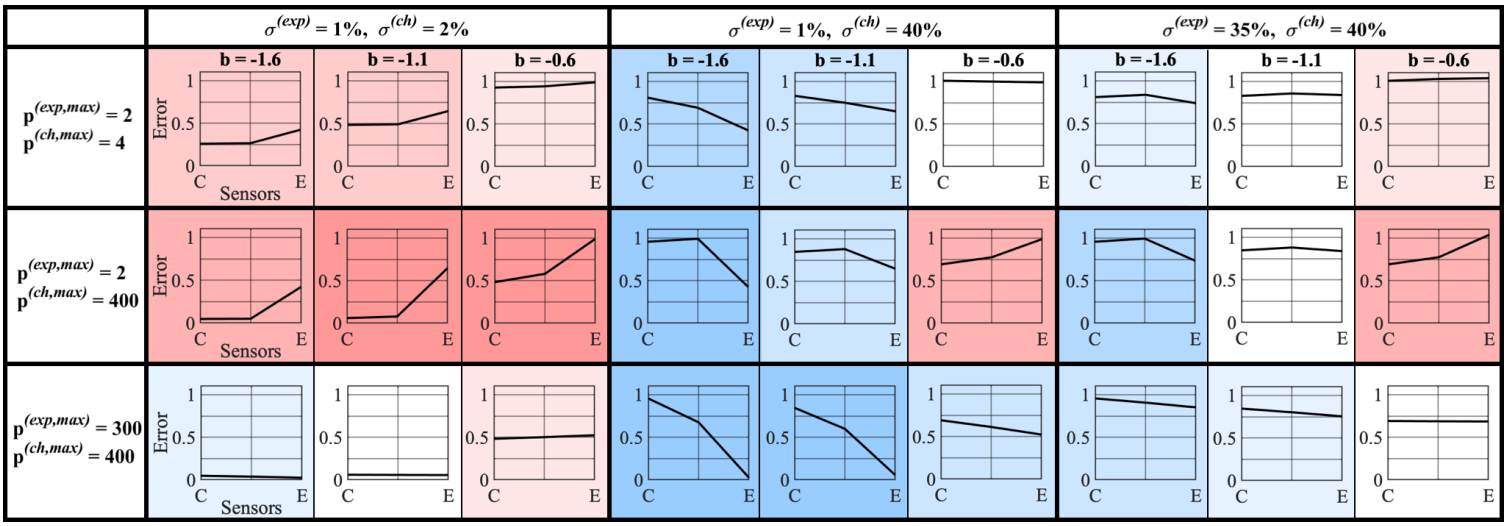

Figure 6.3: Results for placing two different types of sensors in the nine asymptotic regimes of cheap and expensive sensor number and noise level. The plots show reconstruction error versus the number of cheap and expensive sensors, with the first data points (marked as C on the x-axis) using all cheap sensors and the last (E on the x-axis) using all expensive sensors. The plots are color coded based on whether error is lower using all cheap sensors (red), all expensive sensors (blue), or if the results are inconclusive (white, errors are within 2% of each other), as well as shaded from light to dark based on the magnitude of the difference.

plays an important role in the results. The system with $b = -1.6$ is more likely to perform better with a small number of expensive sensors since the system is low rank and does not require a large number of modes or sensors for an accurate reconstruction. Meanwhile, the high-rank data with $b = -0.6$ usually does better with a larger number of cheap sensors, even when those cheap sensors have a high noise level. Finally, the system with medium singular value decay requires a moderate number of sensors and modes for good reconstruction, and is the most likely to have comparable performances with all cheap and all expensive sensors.

In these asymptotic regimes, it is never preferable to have a mix of cheap and expensive sensors, and there are a few cases where a mixture has the worst performance, such as the $b = -1.6$ panel of the middle row and column. Apparently the balance between improving results by adding sensors and improving results by reducing noise is suboptimal in this case.

We note that many of these reconstructions are very poor no matter which sensors are

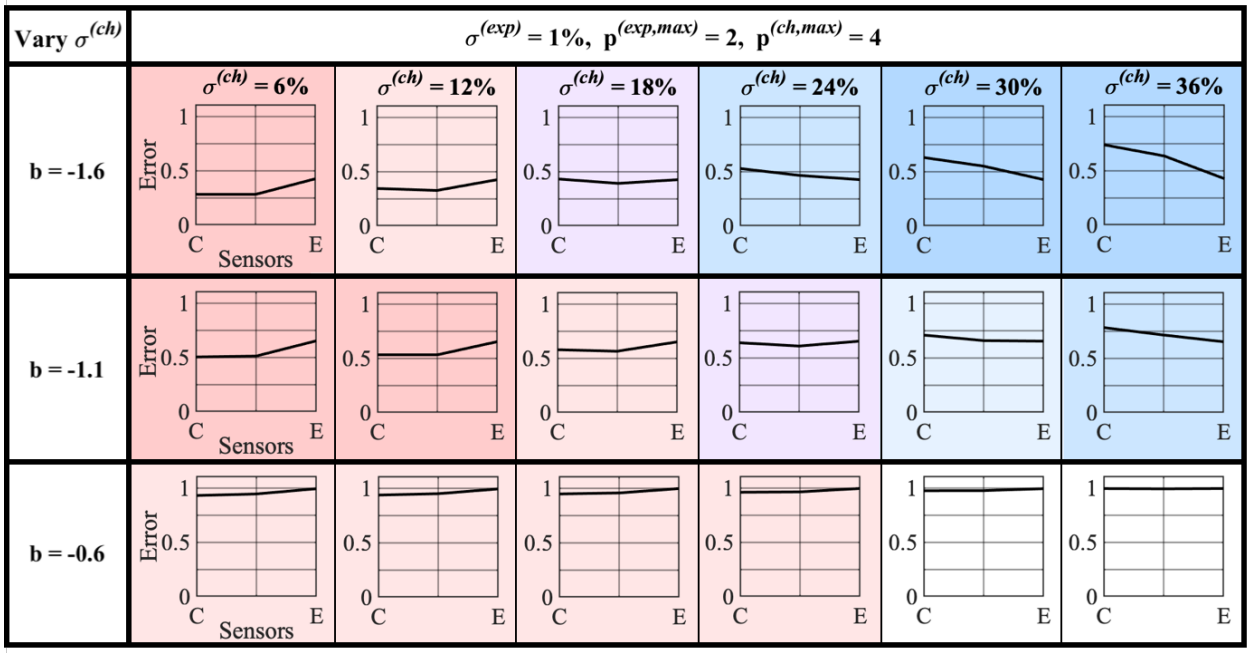

Figure 6.4: Slices across cheap sensor noise level. Each row corresponds to one singular value exponent and each column has a different value of $\sigma^{(ch)}$, beginning at 6% and increasing to 36%. All panels are in the regime of low $\sigma^{(exp)}$, $p^{(exp)}$, and $p^{(ch)}$. As the results transition from favoring all cheap sensors to favoring all expensive sensors, the data sets with $b = -1.6$ and -1.1 pass through a point where it is optimal to have a mix of one expensive sensor and two cheap sensors, shown in purple. The rest of the color scheme is as in Figure 6.3.

used. An eigenface reconstruction with error greater than around 20% begins to look unrecognizable, so the high-noise or low-sensor cases where the reconstruction errors approach or exceed 100% are entirely dominated by noise. In a real-world case like this, the engineer or scientist would need to either purchase a larger number of expensive sensors or design sensors with less noise.

As previously discussed, the parameter space for the multi-fidelity sensor problem is very large, and exploring all cases between the asymptotic regimes shown in Figure 6.3 is prohibitive. However, in Figure 6.4, 6.5, and 6.6, we show a few intermediary results, made by varying one parameter at a time.

Fig. 6.4 shows results for all three singular value exponents, holding $\sigma^{(exp)}$, $p^{(exp,max)}$,

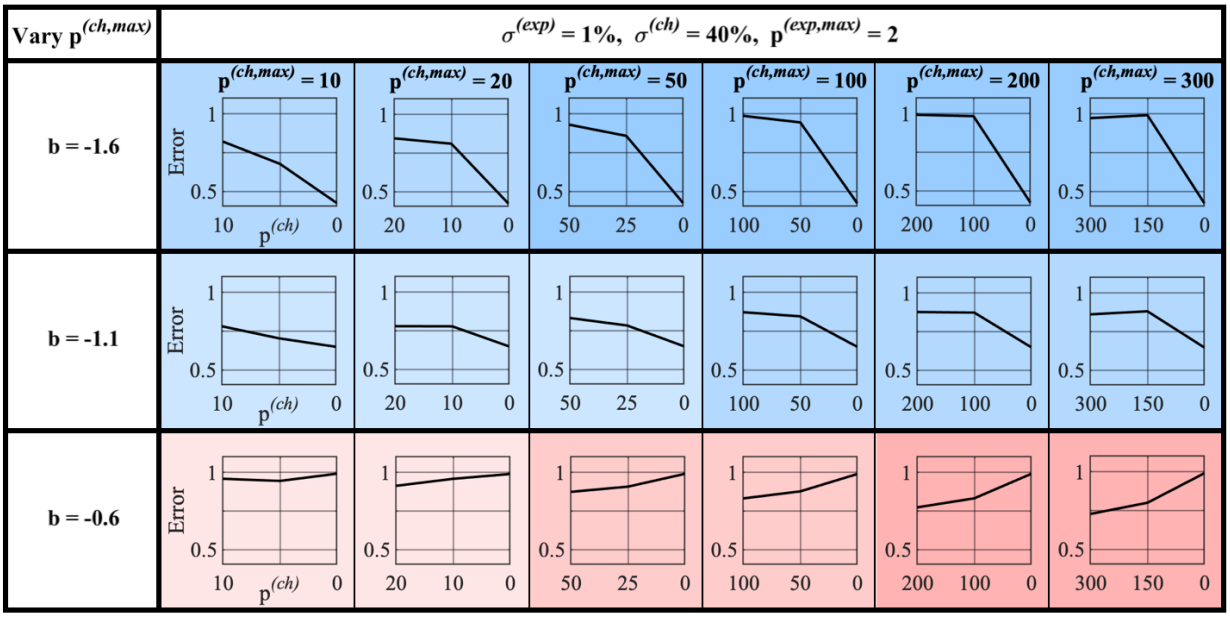

Figure 6.5: Slices across maximum number of cheap sensors. As in Figure 6.4, each row corresponds to a value of b, but here each column has a different maximum number of cheap sensors, increasing from left to right, equivalent to decreasing $c^{(ch)}$. In this case, $p^{(ch)}$ is labeled on the x-axis, but with $p^{(exp,max)} = 2$, $p^{(exp)}$ implicitly increases along the x-axis from zero to two. All panels have $\sigma^{(exp)} = 1\%$ and $\sigma^{(ch)} = 40\%$.

and $p^{(ch,max)}$ constant at low values, and increasing $\sigma^{(ch)}$ from a low to a high value. The data sets with fast and medium singular value decays both transition from favoring all cheap sensors to all expensive sensors, passing through a small window of $\sigma^{(ch)}$ values where a mix of sensor types produces the best results (shown in purple). A mix only outperforms the use of all cheap or all expensive sensors by about 3 or 4%. The data set with slow singular value decay slightly favors the use of all cheap sensors at low values of $\sigma^{(ch)}$, and at higher values of $\sigma^{(ch)}$ it achieves comparable results with either all cheap or all expensive sensors. There is no point in Fig. 6.4 where it is significantly better to use a mix when $b = -0.6$, and since the system has a very high rank, all of the reconstructions are very poor. It is interesting how infrequently a mix of sensor types is optimal, though presumably a wider search over parameter values would locate more conditions where a mix significantly outperforms the

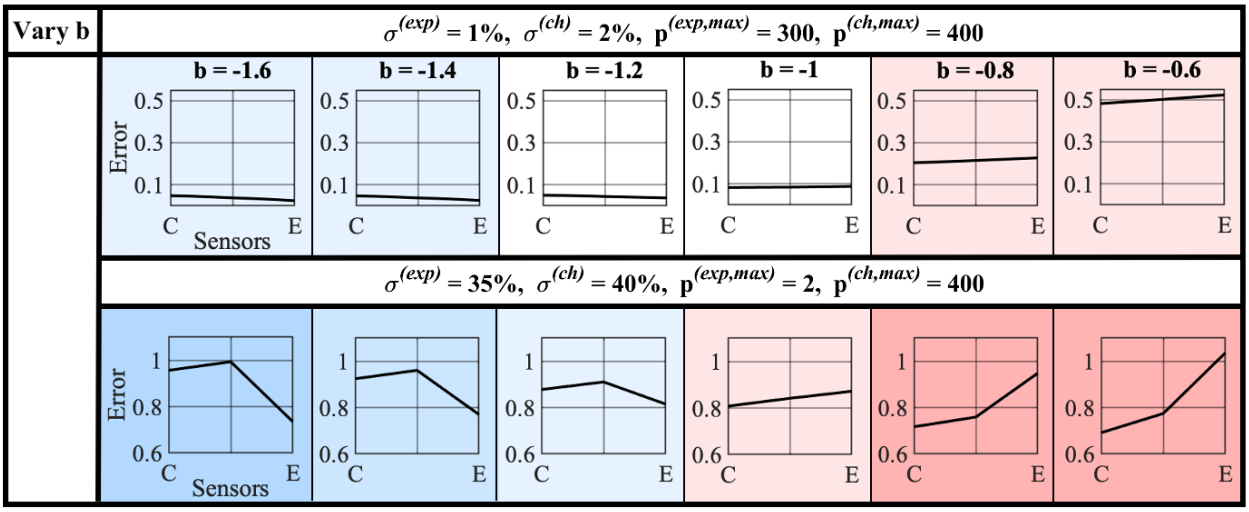

Figure 6.6: Slices across singular value exponent. Each column has a different value of b, increasing from -1.6 to -0.6. The top row is in the regime of low $\sigma^{(ch)}$ and $\sigma^{(exp)}$ and high $p^{(ch)}$ and $p^{(exp)}$. The lower row has high noise values, with low $p^{(exp)}$ and high $p^{(ch)}$.

use of all one type of sensor.

In Fig. 6.5, we vary the maximum number of cheap sensors, with $p^{(exp,max)} = 2$, $\sigma^{(exp)} = 1\%$, and $\sigma^{(ch)} = 40\%$, with results shown for all three values of b. In this case, increasing the maximum number of cheap, noisy sensors intensifies the results seen in this regime in Figure 6.3. For the $b = -1.6$ and -1.1 systems, adding additional, highly noisy sensors decreases the reconstruction quality, and for $p^{(ch,max)} = 300$, a mix of sensor types yields the worst performance. Meanwhile, for the high-rank system, adding sensors improves the reconstruction, even though the cheap sensors have extremely high noise levels. Based on the parameters chosen here, none of the reconstructions are very good (notice the range of the y-axes) for any of the data sets.

Finally, we vary the singular value exponent b for two different sets of the remaining parameters, as shown in Fig. 6.6. In both cases, the systems transition from preferring a small number of expensive sensors to a large number of cheap sensors as b is increased from -1.6 to -0.6. This is reasonable, as the rank of the system increases with b, meaning that

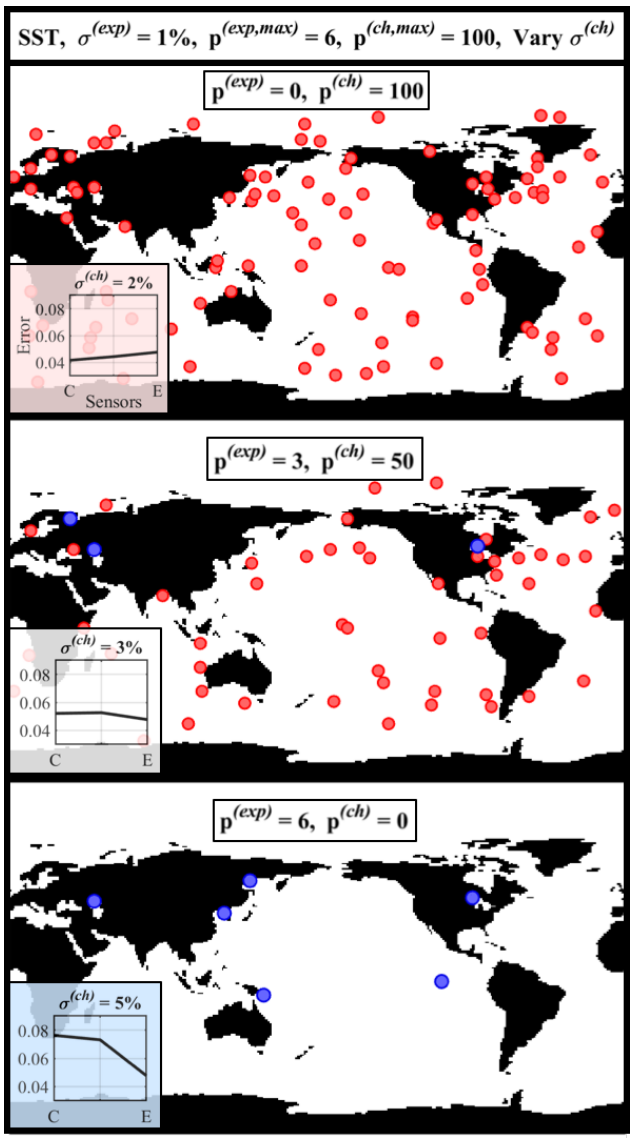

Figure 6.7: Sensor location results for the sea surface temperature data set, selecting up to six expensive sensors with 1% noise level and up to 100 cheap sensors with varying noise levels. The cheap sensors are shown in red and the expensive sensors in blue. The insets give the error versus the number of cheap and expensive sensors, as in previous figures. The trade-off between cheap and expensive sensors, and their combined use, is highly informative for the global monitoring of the sea surface temperature.

more modes and sensors are required to obtain an accurate reconstruction. At no point in Fig. 6.6 is it better to have a mix of sensor types, and when both sensor types have high

noise levels and the system is low rank, a mix gives the poorest results.

To demonstrate that the trends above also apply to real-world data, we consider the NOAA weekly sea surface temperature (SST) data [70–72]. This data set consists of weekly sea surface temperature snapshots from 1990 to 2016, on a 360×180 grid. The system appears to have medium singular value decay, with $b = -1.2$. We select sensors as above, using an SVD basis with $p = 2r$ oversampling in the case of $p > 10$, where noise amplification begins to be significant (when $p < 10$, we set $r = p$). Expensive sensors have 1% noise level and $p^{(exp,max)} = 6$, and we allow up to 100 cheap sensors with noise levels of 2, 3, and 5%. Results are given in Fig. 6.7, where the panels show example sensor placements corresponding to optimum performance given the conditions: 100 cheap sensors when $\sigma^{(ch)}$ is low, and 6 expensive sensors when $\sigma^{(ch)}$ is relatively high. When $\sigma^{(ch)} = 3\%$, the performances are comparable for each distribution of sensors, so for interest we plot the locations of a mix of 3 expensive and 50 cheap sensors. We find that the expensive sensors are more likely to be placed in landlocked seas or lakes, as these measurements are mostly independent from the oceans. The insets show error versus the number of cheap and expensive sensors, as in previous figures.

The reconstructions are highly dependent on the cheap sensor noise level. The system transitions from preferring a large number of cheap sensors to a small number of expensive sensors as $\sigma^{(ch)}$ increases from 2 to just 5%. These reconstructions are of high quality, with errors between about 4 and 8%, which may be the reason why they are so sensitive to noise—the sensors are already resolving details of the snapshot, but with too much additive noise, the fine resolution is lost.

Finally, recall that with a mix of sensors, the expensive sensors were placed at the first QR pivots. However, we also tested 100 other random combinations of placing 3 expensive sensors out of 53 sensors total, and found that remarkably, the average reconstruction errors varied by less than 0.2% as the expensive sensor locations were varied.

All of these figures cover only a small section of the total parameter space for the multi-fidelity sensor selection problem. We have discovered trends in the results, but no analytic

rule for when to choose all cheap, all expensive, or a mix of sensors, and where to place them in the case of a mix. Future work must continue to investigate the influence of sensor noise levels, costs, budget, and system rank on reconstruction quality and the choice of sensor type.

6.4 Conclusions

We consider sparse, multi-fidelity sensor selection for full-state reconstruction in the case of two types of available sensors: (i) low noise with high cost (large signal-to-noise) and (ii) high noise with low cost (low signal-to-noise). The problem and results are complex and nuanced, with a large, non-convex parameter space. Regardless, we provide an initial exploration of a few asymptotic cases of low and high noise levels, and low and high numbers of sensors. Because results are dependent on the rank of the measured data, we construct three artificial data sets with slow, medium, and fast singular value decay. We employ the column-pivoted QR decomposition for sensor placement and find a few general trends for the multi-fidelity sensor selection. Under our chosen asymptotic conditions, it is never better to use a mix of both types of sensors. If both sensors have low noise, it is usually better to use a large number of cheap sensors than a small number of expensive sensors. If the cheap sensors have much higher noise levels than the expensive sensors, then a small number of expensive sensors usually performs better. And if both types of sensor have very high noise levels, reconstructions are generally very poor and which type performs better is rank-dependent: low-rank systems slightly favor a small number of expensive sensors, while high-rank systems do slightly better with a large number of cheap sensors.

Multi-fidelity sensor selection is a highly complex problem, and this chapter is just an initial exploratory study. Even so, to our knowledge it is the first of its kind, and it lays the groundwork for the future development of a complete set of rules for multi-fidelity sensor selection based on cost, noise level, budget, and the singular value spectrum of the data set. This will lead to more efficient sensor arrays with better reconstructions and cost effectiveness.

Chapter 7

DISCUSSION AND FUTURE DIRECTIONS

In this work, we have considered several extensions of a well-known greedy sparse sensor selection method to increase its real-world applicability, with a particular emphasis on sensor cost. We described the column-pivoted QR algorithm and motivated greedy determinant maximization for sensor selection, and then developed an extension of the algorithm that accounts for a heterogeneous cost function on sensor location. This modified algorithm selects locations that are approximately Pareto optimal in cost and reconstruction quality. We considered two common basis choices for modal decompositions, and showed the importance of selecting the optimum number of sensors and modes for each basis. We then applied the cost-constrained QR algorithm to other, dynamically relevant bases, demonstrating that it is important to select the right basis given a system and application. Finally, we turned to the complex problem of multi-fidelity sensor selection, and developed guidelines for choosing cheap or expensive sensors in asymptotic cases of low signal-to-noise sensors with high cost and high signal-to-noise sensors with low cost.

This has not been a complete treatment of all the practical aspects of sensor selection. Certain simplifications we have made include assuming point sensors and linear interpolation maps. These have a wide applicability, but nonlinear bases, such as neural network autoencoders [119], and sensors with a finite range, e.g. [48, 120], would account for many more real-world problems. Autonomous vehicles are just one example of a system that relies heavily on extended imaging sensors [121], and that requires excellent parameter recovery.

One weakness of the cost-constrained column-pivoted QR decomposition advocated here is its lack of performance guarantees. We believe that it is worth systematically exploring both the effect of using a more cost-intensive selection scheme, as in the Gu and Eisenstat

approach [31], and the effect of using performance metrics other than determinant maximization, like the frame potential [44] and mutual information [47, 48], for cost-constrained reconstruction. This may be the subject of future research.

In applying our algorithm to truncated balanced modes for sensor and actuator selection, we found that it may not be preferred for actuator selection, as it does not optimize for the LQR control performance metric. Perhaps there is some modification to the balancing transformation that accounts for the directions of high control cost.

We considered several different bases for sensor selection, but we expect that others will also be effective, and it will be interesting to apply the cost-constrained QR algorithm to more complex systems. In particular fluid flows, such as the flow near a turbulent jet [122], flow near an adjustable flap [3], and control of the flow over an open cavity [123, 124]. Many interesting systems are multiscale in space or time, and we could consider sensor selection for such systems, similar to [91].

As for the multi-fidelity sensor selection problem, we performed only an initial empirical exploration. The ultimate objective is to discover a set of principles for how many of each sensor type to use and where to place them, given the rank of the data, the sensor costs and noise levels, and a set budget. Such principles are difficult to discover except in various asymptotic regimes. Regardless, the work suggests how further studies can be used to reveal more trends, even in non-asymptotic cases. Further extensions could include a weighted or statistical reconstruction method, which could help account for sensor noise, improving reconstructions and perhaps making a mix of sensor types more viable. If a mix of sensor types is preferred, the optimal configuration of cheap and expensive sensors must be determined—should the expensive sensors be placed on the first set of QR pivots, the last set, or in some other configuration entirely? It would also be interesting to consider more than two types of sensors, or sensors that are multi-fidelity in some other way, like modality, bandwidth, or time resolution. We could also improve cost effectiveness by using the cost-constrained QR decomposition. Or we may find that a greedy selection algorithm is not preferred for such a complex multi-parameter problem, and that a convex relaxation

may be more suitable, as in [116, 117].

We note that the sensor cost does not have to be in dollars, but could instead be a measure of the difficulty of installing sensors in certain locations. With this understanding, our cost-constrained sensor selection method could be used to reveal new physics by placing sensors in the optimum locations near totally inaccessible regions. When combined with computer simulations, the measurements could reveal the true conditions inside, for example, a tokamak fusion reactor [125] or a supersonic scramjet [126], both of which have such extreme conditions that they would destroy a sensor placed directly at the most informative locations.

Sensor selection is often treated simplistically mathematically, but all the practicalities of physical sensors, like cost and noise level, make it a highly complex problem. This text treats just a few facets of the practical problem, but even so, it is an important step in bridging the gap between idealized mathematical sensor selection methods and the real world with restrictions like budget and noise. This work combines mathematical principles with real-world restrictions to achieve sensor selections that are nearly optimal in terms of performance, cost, and noise. Moreover, these initial treatments should lead to further developments, yielding better reconstructions as well as cost savings for practitioners looking to deploy networks of sensors.

BIBLIOGRAPHY

[1] J. F. Keithley, *The story of electrical and magnetic measurements: from 500 BC to the 1940s*. John Wiley & Sons, 1999.

[2] G. Antchev, P. Aspell, I. Atanassov, V. Avati, V. Berardi, M. Berretti, M. Bozzo, E. Brucken, A. Buzzo, F. Cafagna, *et al.*, "The TOTEM detector at LHC," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 617, no. 1-3, pp. 62–66, 2010.

[3] J. Taylor and M. N. Glauser, "Towards practical flow sensing and control via POD and LSE based low-dimensional tools," *Journal of Fluids Engineering*, vol. 126, no. 3, pp. 337–345, 2004.

[4] K. Willcox, "Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition," *Computers & Fluids*, vol. 35, no. 2, pp. 208–226, 2006.

[5] S. Chaturantabut and D. C. Sorensen, "Nonlinear model reduction via discrete empirical interpolation," *SIAM Journal on Scientific Computing*, vol. 32, no. 5, pp. 2737–2764, 2010.

[6] K. Lim, "Method for optimal actuator and sensor placement for large flexible structures," *Journal of Guidance, Control, and Dynamics*, vol. 15, no. 1, pp. 49–57, 1992.

[7] S. L. Padula and R. K. Kincaid, "Optimization strategies for sensor and actuator placement," 1999.

[8] P. G. Maghami and S. M. Joshi, "Sensor/actuator placement for flexible space structures," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 2, pp. 345–351, 1993.

[9] K. Hiramoto, H. Doki, and G. Obinata, "Optimal sensor/actuator placement for active vibration control using explicit solution of algebraic Riccati equation," *Journal of Sound and Vibration*, vol. 229, no. 5, pp. 1057–1075, 2000.

[10] X. Yang, D. Venturi, C. Chen, C. Chryssostomidis, and G. E. Karniadakis, "EOF-based constrained sensor placement and field reconstruction from noisy ocean measurements: Application to Nantucket Sound," *Journal of Geophysical Research: Oceans*, vol. 115, no. C12, 2010.

[11] W. Zhang and X. Ma, "Simultaneous fault detection and sensor selection for condition monitoring of wind turbines," *Energies*, vol. 9, no. 4, p. 280, 2016.

[12] J. M. Dolan, G. Podnar, S. Stancliff, E. Lin, J. Hosler, T. Ames, J. Moisan, T. Moisan, J. Higinbotham, and A. Elfes, "Harmful algal bloom characterization via the telesupervised adaptive ocean sensor fleet," 2007.

[13] T. Koshizen, "Improved sensor selection technique by integrating sensor fusion in robot position estimation," *Journal of Intelligent and Robotic Systems*, vol. 29, no. 1, pp. 79–92, 2000.

[14] A. W. Mahoney, T. L. Bruns, P. J. Swaney, and R. J. Webster, "On the inseparable nature of sensor selection, sensor placement, and state estimation for continuum robots or "where to put your sensors and how to use them"," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4472–4478, IEEE, 2016.

[15] E. Samadiani, Y. Joshi, H. Hamann, M. K. Iyengar, S. Kamalsy, and J. Lacey, "Reduced order thermal modeling of data centers via distributed sensor data," *Journal of Heat Transfer*, vol. 134, no. 4, 2012.

[16] L. Liu, S. Wang, D. Liu, Y. Zhang, and Y. Peng, "Entropy-based sensor selection for condition monitoring and prognostics of aircraft engine," *Microelectronics Reliability*, vol. 55, no. 9-10, pp. 2092–2096, 2015.

[17] S.-L. Chua and L. K. Foo, "Sensor selection in smart homes," *Procedia Computer Science*, vol. 69, pp. 116–124, 2015.

[18] K. Manohar, T. Hogan, J. Buttrick, A. G. Banerjee, J. N. Kutz, and S. L. Brunton, "Predicting shim gaps in aircraft assembly with machine learning and sparse sensing," *Journal of manufacturing systems*, vol. 48, pp. 87–95, 2018.

[19] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[20] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[21] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[22] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[23] E. J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[24] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, 2007.

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

[26] G. Berkooz, P. Holmes, and J. L. Lumley, "The proper orthogonal decomposition in the analysis of turbulent flows," *Annual Review of Fluid Mechanics*, vol. 25, no. 1, pp. 539–575, 1993.

[27] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, "An "empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations," *Comptes Rendus Mathematique*, vol. 339, no. 9, pp. 667–672, 2004.

[28] R. Everson and L. Sirovich, "Karhunen–Loeve procedure for gappy data," *Journal of the Optical Society of America A*, vol. 12, no. 8, pp. 1657–1664, 1995.

[29] Y. Zhang and J. G. Bellingham, "An efficient method of selecting ocean observing locations for capturing the leading modes and reconstructing the full field," *Journal of Geophysical Research: Oceans*, vol. 113, no. C4, 2008.

[30] B. Yildirim, C. Chryssostomidis, and G. Karniadakis, "Efficient sensor placement for ocean measurements using low-dimensional concepts," *Ocean Modelling*, vol. 27, no. 3-4, pp. 160–173, 2009.

[31] M. Gu and S. C. Eisenstat, "Efficient algorithms for computing a strong rank-revealing QR factorization," *SIAM Journal on Scientific Computing*, vol. 17, no. 4, pp. 848–869, 1996.

[32] H. Cheng, Z. Gimbutas, P.-G. Martinsson, and V. Rokhlin, "On the compression of low rank matrices," *SIAM Journal on Scientific Computing*, vol. 26, no. 4, pp. 1389–1404, 2005.

[33] C. Li, S. Jegelka, and S. Sra, "Polynomial time algorithms for dual volume sampling," in *Advances in Neural Information Processing Systems*, pp. 5038–5047, 2017.

[34] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 451–462, 2009.

[35] P.-G. Martinsson, V. Rokhlin, and M. Tygert, "On interpolation and integration in finite-dimensional spaces of bounded functions," *Communications in Applied Mathematics and Computational Science*, vol. 1, no. 1, pp. 133–142, 2007.

[36] Z. Drmac and S. Gugercin, "A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions," *SIAM Journal on Scientific Computing*, vol. 38, no. 2, pp. A631–A648, 2016.

[37] E. Liberty, F. Woolfe, P.-G. Martinsson, V. Rokhlin, and M. Tygert, "Randomized algorithms for the low-rank approximation of matrices," *Proceedings of the National Academy of Sciences*, vol. 104, no. 51, pp. 20167–20172, 2007.

[38] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Review*, vol. 53, no. 2, pp. 217–288, 2011.

[39] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functionsI," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[40] M. Coutino, S. P. Chepuri, and G. Leus, "Subset selection for kernel-based signal reconstruction," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4014–4018, IEEE, 2018.

[41] A. Krause and D. Golovin, "Submodular function maximization.," 2014.

[42] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg, "Near-optimal sensor placements: Maximizing information while minimizing communication cost," in *Proceedings of the 5th International Conference on Information Processing in Sensor Networks*, pp. 2–10, ACM, 2006.

[43] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, vol. 7, pp. 1650–1654, 2007.

[44] J. Ranieri, A. Chebira, and M. Vetterli, "Near-optimal sensor placement for linear inverse problems," *IEEE Transactions on Signal Processing*, vol. 62, no. 5, pp. 1135–1146, 2014.

[45] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *49th IEEE conference on decision and control (CDC)*, pp. 2572–2577, IEEE, 2010.

[46] A. Das and D. Kempe, "Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection," *arXiv preprint arXiv:1102.3975*, 2011.

[47] C. Guestrin, A. Krause, and A. P. Singh, "Near-optimal sensor placements in Gaussian processes," in *Proceedings of the 22nd International Conference on Machine Learning*, pp. 265–272, ACM, 2005.

[48] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research*, vol. 9, no. Feb, pp. 235–284, 2008.

[49] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 420–429, ACM, 2007.

[50] R. Iyer and J. Bilmes, "Algorithms for approximate minimization of the difference between submodular functions, with applications," *arXiv preprint arXiv:1207.0560*, 2012.

[51] D. C. Kammer, "Sensor placement for on-orbit modal identification and correlation of large space structures," *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 251–259, 1991.

[52] S. Martínez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," *Automatica*, vol. 42, no. 4, pp. 661–668, 2006.

[53] X. Cheng, D.-Z. Du, L. Wang, and B. Xu, "Relay sensor placement in wireless sensor networks," *Wireless Networks*, vol. 14, no. 3, pp. 347–355, 2008.

[54] T. H. Summers, F. L. Cortesi, and J. Lygeros, "On submodularity and controllability in complex dynamical networks," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 1, pp. 91–101, 2016.

[55] F. L. Cortesi, T. H. Summers, and J. Lygeros, "Submodularity of energy related controllability metrics," in *53rd IEEE Conference on Decision and Control*, pp. 2883–2888, IEEE, 2014.

[56] T. H. Summers and J. Lygeros, "Optimal sensor and actuator placement in complex dynamical networks," *arXiv preprint arXiv:1306.2491*, 2013.

[57] I. Shames and T. H. Summers, "Rigid network design via submodular set function optimization," *IEEE Transactions on Network Science and Engineering*, vol. 2, no. 3, pp. 84–96, 2015.

[58] V. Tzoumas, L. Carlone, G. J. Pappas, and A. Jadbabaie, "Control and sensing co-design," *arXiv preprint arXiv:1802.08376*, 2018.

[59] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.

[60] E. Clark, J. N. Kutz, and S. L. Brunton, "Sensor selection with cost constraints for dynamically relevant bases," *IEEE Sensors Journal*, 2020.

[61] P. Businger and G. H. Golub, "Linear least squares solutions by Householder transformations," *Numerische Mathematik*, vol. 7, no. 3, pp. 269–276, 1965.

[62] A. S. Householder, "Unitary triangularization of a nonsymmetric matrix," *Journal of the ACM (JACM)*, vol. 5, no. 4, pp. 339–342, 1958.

[63] P. Y. Papalambros and D. J. Wilde, *Principles of optimal design: Modeling and computation*. Cambridge university press, 2000.

[64] B. W. Brunton, S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Sparse sensor placement optimization for classification," *SIAM Journal on Applied Mathematics*, vol. 76, no. 5, pp. 2099–2122, 2016.

[65] L. Clemmensen, T. Hastie, D. Witten, and B. Ersbøll, "Sparse discriminant analysis," *Technometrics*, vol. 53, no. 4, pp. 406–413, 2011.

[66] D. Cai, X. He, Y. Hu, J. Han, and T. Huang, "Learning a spatially smooth subspace for face recognition," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7, IEEE, 2007.

[67] D. Cai, X. He, and J. Han, "Spectral regression for efficient regularized subspace learning," in *2007 IEEE 11th International Conference on Computer Vision*, pp. 1–8, IEEE, 2007.

[68] D. Cai, X. He, J. Han, and H.-J. Zhang, "Orthogonal laplacianfaces for face recognition," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3608–3614, 2006.

[69] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.

[70] "NOAA optimal interpolation (OI) sea surface temperature (SST) v2."

[71] V. Banzon, T. M. Smith, T. M. Chin, C. Liu, and W. Hankins, "A long-term record of blended satellite and in situ sea-surface temperature for climate monitoring, modeling and environmental studies," *Earth System Science Data*, vol. 8, no. 1, pp. 165–176, 2016.

[72] R. W. Reynolds, T. M. Smith, C. Liu, D. B. Chelton, K. S. Casey, and M. G. Schlax, "Daily high-resolution-blended analyses for sea surface temperature," *Journal of Climate*, vol. 20, no. 22, pp. 5473–5496, 2007.

[73] K. Taira and T. Colonius, "The immersed boundary method: A projection approach," *Journal of Computational Physics*, vol. 225, no. 2, pp. 2118–2137, 2007.

[74] T. Colonius and K. Taira, "A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 25-28, pp. 2131–2146, 2008.

[75] B. W. Brunton, S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Optimal sensor placement and enhanced sparsity for classification," *arXiv preprint arXiv:1310.4217*, 2013.

[76] M. Gavish and D. L. Donoho, "The optimal hard threshold for singular values is $4/\sqrt{3}$," *IEEE Transactions on Information Theory*, vol. 60, no. 8, pp. 5040–5053, 2014.

[77] B. Peherstorfer, Z. Drmač, and S. Gugercin, "Stabilizing discrete empirical interpolation via randomized and deterministic oversampling," *arXiv preprint arXiv:1808.10473*, 2018.

[78] E. Clark, T. Askham, S. L. Brunton, and J. N. Kutz, "Greedy sensor placement with cost constraints," *IEEE Sensors Journal*, vol. 19, no. 7, pp. 2642–2656, 2018.

[79] J. Argaud, B. Bouriquet, H. Gong, Y. Maday, and O. Mula, "Stabilization of (G) EIM in presence of measurement noise: Application to nuclear reactor physics," in *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2016*, pp. 133–145, Springer, 2017.

[80] R. Zimmermann and K. Willcox, "An accelerated greedy missing point estimation procedure," *SIAM Journal on Scientific Computing*, vol. 38, no. 5, pp. A2827–A2850, 2016.

[81] N. B. Erichson, S. Voronin, S. L. Brunton, and J. N. Kutz, "Randomized matrix decompositions using R," *arXiv preprint arXiv:1608.02148*, 2016.

[82] K. Taira, S. L. Brunton, S. T. Dawson, C. W. Rowley, T. Colonius, B. J. McKeon, O. T. Schmidt, S. Gordeyev, V. Theofilis, and L. S. Ukeiley, "Modal analysis of fluid flows: An overview," *AIAA Journal*, pp. 4013–4041, 2017.

[83] K. Taira, M. S. Hemati, S. L. Brunton, Y. Sun, K. Duraisamy, S. Bagheri, S. T. Dawson, and C.-A. Yeh, "Modal analysis of fluid flows: Applications and outlook," *AIAA Journal*, vol. 58, no. 3, pp. 998–1022, 2020.

[84] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2019.

[85] G. H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," in *Linear Algebra*, pp. 134–151, Springer, 1971.

[86] B. Moore, "Principal component analysis in linear systems: Controllability, observability, and model reduction," *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 17–32, 1981.

[87] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, pp. 5–28, 2010.

[88] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *Journal of Fluid Mechanics*, vol. 641, pp. 115–127, 2009.

[89] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, "On dynamic mode decomposition: Theory and applications," *Journal of Computational Dynamics*, vol. 1, no. 2, pp. 391–421, 2014.

[90] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic mode decomposition: Data-driven modeling of complex systems*. SIAM, 2016.

[91] K. Manohar, E. Kaiser, S. L. Brunton, and J. N. Kutz, "Optimized sampling for multiscale dynamics," *Multiscale Modeling & Simulation*, vol. 17, no. 1, pp. 117–136, 2019.

[92] K. Manohar, J. N. Kutz, and S. L. Brunton, "Optimal sensor and actuator placement using balanced model reduction," *arXiv preprint arXiv:1812.01574*, 2018.

[93] N. K. Dhingra, M. R. Jovanović, and Z.-Q. Luo, "An ADMM algorithm for optimal sensor and actuator selection," in *53rd IEEE Conference on Decision and Control*, pp. 4039–4044, IEEE, 2014.

[94] U. Münz, M. Pfister, and P. Wolfrum, "Sensor and actuator placement for linear systems based on H_2 and H_∞ optimization," *IEEE Transactions on Automatic Control*, vol. 59, no. 11, pp. 2984–2989, 2014.

[95] T. H. Summers and J. Lygeros, "Optimal sensor and actuator placement in complex dynamical networks," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 3784–3789, 2014.

[96] T. H. Summers, F. L. Cortesi, and J. Lygeros, "On submodularity and controllability in complex dynamical networks," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 1, pp. 91–101, 2015.

[97] V. Tzoumas, M. A. Rahimian, G. J. Pappas, and A. Jadbabaie, "Minimal actuator placement with bounds on control effort," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 1, pp. 67–78, 2015.

[98] T. Nestorović and M. Trajkov, "Optimal actuator and sensor placement based on balanced reduced models," *Mechanical Systems and Signal Processing*, vol. 36, no. 2, pp. 271–289, 2013.

[99] K. Willcox and J. Peraire, "Balanced model reduction via the proper orthogonal decomposition," *AIAA journal*, vol. 40, no. 11, pp. 2323–2330, 2002.

[100] C. W. Rowley, "Model reduction for fluids, using balanced proper orthogonal decomposition," *International Journal of Bifurcation and Chaos*, vol. 15, no. 03, pp. 997–1013, 2005.

[101] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Dynamic mode decomposition with control," *SIAM Journal on Applied Dynamical Systems*, vol. 15, no. 1, pp. 142–161, 2016.

[102] B. Kramer, P. Grover, P. Boufounos, S. Nabi, and M. Benosman, "Sparse sensing and DMD-based identification of flow regimes and bifurcations in complex flows," *SIAM Journal on Applied Dynamical Systems*, vol. 16, no. 2, pp. 1164–1196, 2017.

[103] I. Mezić, "Analysis of fluid flows via spectral properties of the Koopman operator," *Annual Review of Fluid Mechanics*, vol. 45, pp. 357–378, 2013.

[104] M. R. Jovanović, P. J. Schmid, and J. W. Nichols, "Sparsity-promoting dynamic mode decomposition," *Physics of Fluids*, vol. 26, no. 2, p. 024103, 2014.

[105] S. L. Brunton, J. L. Proctor, J. H. Tu, and J. N. Kutz, "Compressed sensing and dynamic mode decomposition," *Journal of Computational Dynamics*, vol. 2, no. 2, 2015.

[106] F. Guéniat, L. Mathelin, and L. R. Pastur, "A dynamic mode decomposition approach for large and arbitrarily sampled systems," *Physics of Fluids*, vol. 27, no. 2, p. 025113, 2015.

[107] J. H. Tu, C. W. Rowley, J. N. Kutz, and J. K. Shang, "Spectral analysis of fluid flows using sub-Nyquist-rate PIV data," *Experiments in Fluids*, vol. 55, no. 9, p. 1805, 2014.

[108] T. Askham and J. N. Kutz, "Variable projection methods for an optimized dynamic mode decomposition," *SIAM Journal on Applied Dynamical Systems*, vol. 17, no. 1, pp. 380–416, 2018.

[109] D. Lahat, T. Adali, and C. Jutten, "Multimodal data fusion: An overview of methods, challenges, and prospects," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1449–1477, 2015.

[110] C. Zhang and Y. Xu, "Optimal multi-type sensor placement for response and excitation reconstruction," *Journal of Sound and Vibration*, vol. 360, pp. 112–128, 2016.

[111] S. Zhu, X.-H. Zhang, Y.-L. Xu, and S. Zhan, "Multi-type sensor placement for multi-scale response reconstruction," *Advances in Structural Engineering*, vol. 16, no. 10, pp. 1779–1797, 2013.

[112] X. Zhang, Y. Xu, and S. Zhan, "Optimal locations of a multi-type sensor system for structural health monitoring," in *Proceedings Of The 8th International Conference On Structural Dynamics*, 2011.

[113] J.-F. Lin, Y.-L. Xu, and S.-S. Law, "Structural damage detection-oriented multi-type sensor placement with multi-objective optimization," *Journal of Sound and Vibration*, vol. 422, pp. 568–589, 2018.

[114] R. N. Soman, T. Onoufrioua, M. A. Kyriakidesb, R. A. Votsisc, and C. Z. Chrysostomou, "Multi-type, multi-sensor placement optimization for structural health monitoring of long span bridges," *Smart Structures and Systems*, vol. 14, no. 1, pp. 55–70, 2014.

[115] D. C. Kammer, "Effects of noise on sensor placement for on-orbit modal identification of large space structures," *Journal of Dynamic Systems, Measurement and Control*, vol. 114, no. 3, pp. 436–443, 1992.

[116] O. M. Bushnaq, T. Y. Al-Naffouri, S. P. Chepuri, and G. Leus, "Joint sensor placement and power rating selection in energy harvesting wireless sensor networks," in *2017 25th European Signal Processing Conference (EUSIPCO)*, pp. 2423–2427, IEEE, 2017.

[117] O. M. Bushnaq, A. Chaaban, S. P. Chepuri, G. Leus, and T. Y. Al-Naffouri, "Sensor placement and resource allocation for energy harvesting IoT networks," *Digital Signal Processing*, p. 102659, 2020.

[118] M. Shirazi and A. Vosoughi, "On Bayesian Fisher information maximization for distributed vector estimation," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, no. 4, pp. 628–645, 2019.

[119] K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton, "Data-driven discovery of coordinates and governing equations," *Proceedings of the National Academy of Sciences*, vol. 116, no. 45, pp. 22445–22451, 2019.

[120] D. S. Hochbaum and W. Maass, "Approximation schemes for covering and packing problems in image processing and vlsi," *Journal of the ACM (JACM)*, vol. 32, no. 1, pp. 130–136, 1985.

[121] C. Ilas, "Electronic sensing technologies for autonomous ground vehicles: A review," in *2013 8TH INTERNATIONAL SYMPOSIUM ON ADVANCED TOPICS IN ELECTRICAL ENGINEERING (ATEE)*, pp. 1–6, IEEE, 2013.

[122] R. E. Arndt, D. Long, and M. N. Glauser, "The proper orthogonal decomposition of pressure fluctuations surrounding a turbulent jet," *Journal of Fluid Mechanics*, vol. 340, pp. 1–33, 1997.

[123] L. Cattafesta, F. Alvi, D. Williams, and C. Rowley, "Review of active control of flow-induced cavity oscillations," in *33rd AIAA Fluid Dynamics Conference and Exhibit*, p. 3567, 2003.

[124] C. W. Rowley and D. R. Williams, "Dynamics and control of high-Reynolds-number flow over open cavities," *Annual Review of Fluid Mechanics*, vol. 38, pp. 251–276, 2006.

[125] M. A. Beer, G. Hammett, G. Rewoldt, E. Synakowski, M. Zarnstorff, and W. Dorland, "Gyrofluid simulations of turbulence suppression in reversed-shear experiments on the Tokamak Fusion Test Reactor," *Physics of Plasmas*, vol. 4, no. 5, pp. 1792–1799, 1997.

[126] E. T. Curran, "Scramjet engines: The first forty years," *Journal of Propulsion and Power*, vol. 17, no. 6, pp. 1138–1148, 2001.