

SPARSE DISCRETE EMPIRICAL INTERPOLATION METHOD: STATE ESTIMATION FROM FEW SENSORS*

MOHAMMAD FARAZMAND†

Abstract. The discrete empirical interpolation method (DEIM) estimates a function from its incomplete pointwise measurements. Unfortunately, DEIM suffers large interpolation errors when few measurements are available. Here, we introduce Sparse DEIM (S-DEIM) for accurately estimating a function even when very few measurements are available. To this end, S-DEIM leverages a kernel vector which has been neglected in previous DEIM-based methods. We derive theoretical error estimates for S-DEIM, showing its relatively small error when an optimal kernel vector is used. When the function is generated by a continuous-time dynamical system, we propose a data assimilation algorithm which approximates the optimal kernel vector using sparse observational time series. We prove that, under certain conditions, data assimilated S-DEIM converges exponentially fast towards the true state. We demonstrate the efficacy of our method on two numerical examples.

Key words. empirical interpolation, compressed sensing, proper orthogonal decomposition, data assimilation, dynamical systems

MSC codes. 65D05, 65L09, 45Q05, 94A12

DOI. 10.1137/24M1636344

1. Introduction. Estimating the full state of a system from partial observations is needed in many fields such as control theory, fluid mechanics, imaging, meteorology, and oceanography [29]. Methods that seek to accomplish this task go by various names, such as state estimation [11], flow reconstruction [38], compressed sensing [13], and optimal experimental design [33].

Among existing methods, the Discrete Empirical Interpolation Method (DEIM) stands out for its simplicity and interpretability [10]. DEIM, and its later variant Q-DEIM [14], use pointwise measurements of the state of the system to reconstruct the full state as a linear combination of prescribed basis functions (or modes). As we review in section 1.1, (Q-)DEIM first approximates the optimal sensor placement for gathering pointwise observations. Given this data, it then determines the optimal expansion coefficients in the prescribed basis in order to estimate the full state. Our contributions here concern the latter step; we use the Q-DEIM algorithm [14] for sensor placement.

Numerical evidence in [12] suggests that (Q-)DEIM performs best when the number of sensors are greater than or equal to the number of modes. However, in the regime where the number of sensors are smaller than the number of modes, (Q-)DEIM can incur large reconstruction errors. Unfortunately, one frequently encounters this regime in practice since sensors can be quite expensive and furthermore their deployment in large numbers may face practical obstacles [12]. On the other hand, modes

*Submitted to the journal's Numerical Algorithms for Scientific Computing section February 1, 2024; accepted for publication (in revised form) August 30, 2024; published electronically December 3, 2024. The U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes. Copyright is owned by SIAM to the extent not limited by these rights.

<https://doi.org/10.1137/24M1636344>

Funding: This work was supported by the National Science Foundation, the Algorithms for Threat Detection (ATD) program, through the award DMS-2220548.

†Department of Mathematics, North Carolina State University, Raleigh, NC 27695-8205 USA (farazmand@ncsu.edu).

or basis functions are either prescribed analytically (e.g., polynomial interpolation) or are obtained numerically (e.g., via proper orthogonal decomposition). As such, the number of modes can be increased arbitrarily as needed.

This paper addresses the latter regime where the number of sensors is smaller than the number of modes. We first introduce *Sparse DEIM* (S-DEIM) by making use of the kernel vector which has been neglected in previous DEIM-based methods. Our error estimates show that certain values of this kernel vector result in much more accurate state estimation compared to (Q-)DEIM. However, determining the optimal kernel vector from observations is nontrivial. In the case where the state is generated by a continuous-time dynamical system, we propose a data assimilation method which approximates the optimal kernel vector from observational time series. Therefore, our contributions can be summarized as follows:

1. Sparse DEIM: We introduce S-DEIM for state reconstruction when the number of sensors is smaller than the number of modes.
2. Error estimates: We derive a number of error estimates for S-DEIM which quantify its approximation accuracy and the optimal choice of the kernel vector.
3. Ruling out two-stage state estimation: We prove that a two-stage state estimation using S-DEIM, although seemingly promising, fails to improve the (Q-)DEIM reconstruction.
4. Data assimilation with S-DEIM: In the special case, where the state is generated by a dynamical system, we propose a data assimilation method for estimating the optimal kernel vector in S-DEIM. Under certain assumptions, we prove that our method converges exponentially fast towards the true state.

1.1. Related work. Due to its broad applications, a plethora of methods have been developed for state reconstruction from sparse measurements. Broadly speaking, these methods can be divided into two categories. One category of methods is deterministic and relies on solving a least squares problem; DEIM belongs to this category. The second category is based on Bayesian inference and yields a probabilistic reconstruction [3, 22, 25]. Furthermore, recent years have seen a surge of methods based on deep learning [1, 6] which overcome some drawbacks [31] of linear methods such as DEIM. In this section, we only review previous work on (Q-)DEIM and refer to [8, 17] for a broader review of other methods.

DEIM was originally proposed by Chaturantabut and Sorensen [10] for reduced-order modeling of dynamical systems generated by nonlinear differential equations. Later, it became clear that the same methodology can also be used for estimating the full state of the system from its sparse sensor measurements [29].

The original work of Chaturantabut and Sorensen [10] also proposed a method for optimal sensor placement. However, this method was not invariant under reordering of the basis functions. Drmač and Gugercin [14] proposed Q-DEIM to address this issue. Q-DEIM uses QR factorization with column pivoting to approximate the optimal placement of the sensors. Furthermore, the theoretical upper bound for Q-DEIM error is sharper than the one derived in [10]. There are several other generalizations of DEIM, including its application to weighted inner product spaces [15], tensor-based DEIM [17, 24], DEIM for classification [18], and its generalization for state reconstruction on manifolds [30] as opposed to a linear subspace.

(Q-)DEIM is well-known to be sensitive to measurement error. Peherstorfer, Drmač, and Gugercin [32] carry out a probabilistic analysis to quantify the reconstruction error of DEIM when the observations are noisy. A similar analysis, in the

context of D-optimal design, is carried out in [23, 29]. Callaham, Maeda, and Brunton [8] propose several methods which regularize the underlying DEIM optimization problem in order to decrease its sensitivity to noise. Unlike DEIM, their modified optimization problems do not necessarily have a closed-form solution. As a result, their computational cost can be significantly higher than that of DEIM.

DEIM was originally developed under the assumption that the number of sensors n is equal to the number of modes m used in the linear expansion [10]. However, DEIM can easily be generalized to the case where the number of sensor and modes are different ($n \neq m$). Numerical evidence suggests that the (Q-)DEIM reconstruction is most accurate when the number of sensors is greater than or equal to the number of modes ($n \geq m$) [12]. This regression regime has been studied extensively. In fact, when $n > m$, DEIM coincides with gappy proper orthogonal decomposition (POD) which was discovered much earlier [2, 7, 16, 37]. Peherstorfer, Drmač, and Gugercin [32] showed that gappy POD is more robust to observational noise compared to DEIM with $n = m$. Recently, Lauzon et al. [26] proposed a point selection algorithm for optimal placement of additional sensors in gappy POD (also see [34]).

The opposite regime, where fewer sensors are available compared to modes ($n < m$), has received relatively little attention. As we mentioned earlier, this regime arises frequently in practice since procurement and deployment of sensors can be quite expensive. The present work develops the method of S-DEIM to address this regime: fewer sensors than modes.

1.2. Outline. The remainder of this paper is organized as follows. In section 2, we discuss the problem set-up and review some relevant aspects of (Q-)DEIM. In section 3, we present our S-DEIM method and its error analysis. In section 4, we present a computable method for S-DEIM in the case where the states are generated by a continuous-time dynamical system. Section 5 contains our numerical examples. Our concluding remarks are presented in section 6.

2. Preliminaries and set-up. We consider bounded functions $u : \Omega \rightarrow \mathbb{R}, \mathbf{x} \mapsto u(\mathbf{x})$, where $\Omega \subset \mathbb{R}^d$ is the spatial domain over which the function u is defined. We assume that the function u belongs to a Banach space \mathcal{U} . Consider a complete basis $\{\phi_i\}_{i=1}^\infty$ in \mathcal{U} , so that

$$(2.1) \quad \lim_{m \rightarrow \infty} \sum_{i=1}^m c_i \phi_i(\mathbf{x}) = u(\mathbf{x})$$

for appropriate coefficients $c_i \in \mathbb{R}$, where the convergence is uniform in the norm $\|\cdot\|_{\mathcal{U}}$ associated with the Banach space \mathcal{U} .

Consider a finite truncation of the series in (2.1),

$$(2.2) \quad \tilde{u}(\mathbf{x}) = \sum_{i=1}^m c_i \phi_i(\mathbf{x}),$$

for some $m > 0$. We would like to find the coefficients c_i such that $\tilde{u} \simeq u$, where the approximation is understood in an appropriate sense. In order to find the coefficients c_i some additional information about the function u is needed. We assume that the function values are measured at a sparse set of distinct points $\mathcal{S} := \{\mathbf{s}_i\}_{i=1}^n \subset \Omega$. In practice, these measurements are obtained by n sensors located at the points \mathbf{s}_i . Given the measurements $u(\mathbf{s}_i)$, we would like to reconstruct the function u over the entire domain Ω .

Therefore, the problem statement can be informally summarized as follows: Given the basis functions $\{\phi_1(\mathbf{x}), \dots, \phi_m(\mathbf{x})\}$ and the measurements $\{u(\mathbf{s}_1), \dots, u(\mathbf{s}_n)\}$, find the coefficients c_i in (2.2) so that $\tilde{u} \simeq u$.

2.1. Finite-dimensional discretization. In practice, it is often sufficient to know the function u over a dense enough grid $\mathcal{G} := \{\mathbf{x}_i\}_{i=1}^N \subset \Omega$. We assume that N is large enough so that $u(\mathbf{x}_i)$, $i = 1, 2, \dots, N$, amounts to a high-resolution discretization of the function u . This allows us to transform the problem from the infinite-dimensional function space to a finite-dimensional setting. We stack the function values over the high-resolution grid \mathcal{G} into a vector

$$(2.3) \quad \mathbf{u} = [u(\mathbf{x}_1), u(\mathbf{x}_2), \dots, u(\mathbf{x}_N)]^\top \in \mathbb{R}^N.$$

Note, however, that the function values are not known over the entire high-resolution grid \mathcal{G} . Instead of interpolating the function u over the entire domain Ω , we would like to find its values over the grid \mathcal{G} .

Similarly, we discretize the basis functions ϕ_i and define

$$(2.4) \quad \phi_i = [\phi_i(\mathbf{x}_1), \phi_i(\mathbf{x}_2), \dots, \phi_i(\mathbf{x}_N)]^\top \in \mathbb{R}^N, \quad i = 1, 2, \dots, m.$$

We also define the basis matrix $\Phi_m = [\phi_1 | \phi_2 | \dots | \phi_m] \in \mathbb{R}^{N \times m}$. We assume that the basis vectors are orthonormalized, so that $\Phi_m^\top \Phi_m = \mathbb{I}_m$. Finally, approximation (2.2) is discretized to define

$$(2.5) \quad \tilde{\mathbf{u}} = [\tilde{u}(\mathbf{x}_1), \tilde{u}(\mathbf{x}_2), \dots, \tilde{u}(\mathbf{x}_N)]^\top \in \mathbb{R}^N,$$

so that $\tilde{\mathbf{u}} = \Phi \mathbf{c}$, where $\mathbf{c} = [c_1, c_2, \dots, c_m]^\top \in \mathbb{R}^m$ is the vector of unknown coefficients.

Recall that we seek to infer the coefficients \mathbf{c} from the observational data $\{u(\mathbf{s}_1), u(\mathbf{s}_2), \dots, u(\mathbf{s}_n)\}$ which are obtained at the sensor locations $\mathcal{S} = \{\mathbf{s}_k\}_{k=1}^n \subset \mathcal{G}$. We define the vector of observations $\mathbf{y} \in \mathbb{R}^n$ as

$$(2.6) \quad \mathbf{y} = [u(\mathbf{s}_1), u(\mathbf{s}_2), \dots, u(\mathbf{s}_n)]^\top + \boldsymbol{\varepsilon},$$

where $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ is a vector of errors accounting for observational noise. In the following, we only consider the ideal case where the observational noise is negligible and set $\boldsymbol{\varepsilon} = \mathbf{0}$. Nonetheless, in section 5 where our numerical results are presented, we study the effects of noise empirically.

It is convenient to envision the sensor measurements as a linear map from the full state \mathbf{u} to the observations \mathbf{y} . To this end, we define the selection operator $S_n : \mathbb{R}^N \rightarrow \mathbb{R}^n$ which is an $N \times n$ matrix whose columns are a subset of the standard basis $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_N\}$ on \mathbb{R}^N . More specifically, let $\{i_1, i_2, \dots, i_n\}$ denote a subset of the index set $\{1, 2, \dots, N\}$ such that $\mathbf{x}_{i_k} = \mathbf{s}_k$ for $k = 1, 2, \dots, n$. In other words, the distinct indices i_k are chosen so that \mathbf{x}_{i_k} coincides with the sensor locations \mathbf{s}_k . Consequently, defining the *selection matrix* $S_n = [\mathbf{e}_{i_1} | \dots | \mathbf{e}_{i_n}]$, we have

$$(2.7) \quad \mathbf{y} = S_n^\top \mathbf{u}.$$

Therefore, $S_n^\top : \mathbb{R}^N \rightarrow \mathbb{R}^n$ returns the smaller vector \mathbf{y} whose entries coincide with the entries of the full state vector \mathbf{u} at the sensor locations $\mathbf{x}_{i_k} = \mathbf{s}_k$. Conversely, $S_n \mathbf{y}$ returns a vector in \mathbb{R}^N whose entries are zero except the ones corresponding to the sensor locations. In principle, $S_n \mathbf{y}$ is a reconstruction of the full state; however, this reconstruction is very inaccurate since it zeros out all the entries where no sensors

TABLE 1
The quantities and terminology used in this paper.

Symbol	Description
N	High-fidelity resolution
n	Number of sensors
m	Number of modes (basis functions)
$\mathbf{u} \in \mathbb{R}^N$	Unknown state
$\mathbf{y} \in \mathbb{R}^n$	Known observations
$S_n \in \mathbb{R}^{N \times n}$	Selection matrix, $\mathbf{y} = S_n^\top \mathbf{u}$
$\Phi_m = [\phi_1 \dots \phi_m] \in \mathbb{R}^{N \times m}$	Basis matrix
$\hat{\mathbf{u}} \in \mathbb{R}^N$	Reconstructed state
$\hat{\mathbf{u}} \in \mathbb{R}^N$	Orthogonal reconstruction, $\hat{\mathbf{u}} = \Phi_m \Phi_m^\top \mathbf{u}$
$\mathbf{c} \in \mathbb{R}^m$	Expansion coefficients, $\hat{\mathbf{u}} = \Phi_m \mathbf{c}$
$\mathbb{D}_{m,n} = \Phi_m (S_n^\top \Phi_m)^\dagger S_n^\top \in \mathbb{R}^{N \times N}$	DEIM operator

are present. The selection matrix S_n plays an important role in DEIM. In particular, the optimal sensor placement can be equivalently expressed in terms of the selection matrix, as we describe in section 2.2. In the following, we often omit the subscripts m and n , unless they are necessary for clarity. The quantities and terminology that are used repeatedly in this paper are summarized in Table 1.

2.2. Vanilla DEIM. We refer to DEIM and Q-DEIM methods [10, 14] as vanilla DEIM to distinguish them from S-DEIM introduced in section 3. Here, we first review vanilla DEIM for the special case where the number of sensors and modes are equal, $n = m$ (section 2.2.1). This is the setting in which vanilla DEIM is used most often. Towards the end of this section, we discuss the implications of using vanilla DEIM with fewer sensors than modes, $n < m$ (section 2.2.2). DEIM was originally derived in a reduced-order modeling framework. Here, we present a different, but equivalent, derivation which is more natural for interpolation problems.

2.2.1. Equal number of sensors and modes. One can seek the optimal coefficients \mathbf{c} by solving the optimization problem,

$$(2.8) \quad \min_{\mathbf{c} \in \mathbb{R}^m} \|\Phi \mathbf{c} - \mathbf{u}\|^2,$$

where $\|\cdot\|$ denotes the Euclidean 2-norm. This problem has a unique solution $\mathbf{c} = \Phi^\top \mathbf{u}$, leading to the reconstruction $\hat{\mathbf{u}} = \Phi \Phi^\top \mathbf{u}$ which is the nearest point in $\mathcal{R}[\Phi]$ (range of Φ) to the state \mathbf{u} . This motivates the following definition.

DEFINITION 1 (orthogonal reconstruction). *For a basis matrix $\Phi \in \mathbb{R}^{N \times m}$ and a vector $\mathbf{u} \in \mathbb{R}^N$, we refer to the orthogonal projection $\hat{\mathbf{u}} = \Phi \Phi^\top \mathbf{u}$ as the orthogonal reconstruction of \mathbf{u} . We refer to the corresponding error $\mathcal{E}_m(\mathbf{u}) = \|\hat{\mathbf{u}} - \mathbf{u}\|$ as the truncation error.*

The orthogonal reconstruction $\hat{\mathbf{u}}$ is not computable from observations since it requires the knowledge of the entire state \mathbf{u} . Note that the optimization problem (2.8) does not use the observational data \mathbf{y} . One can incorporate the observational data by adding them as constraints to the optimization problem, and solve

$$(2.9a) \quad \min_{\mathbf{c} \in \mathbb{R}^m} \|\Phi \mathbf{c} - \mathbf{u}\|^2,$$

$$(2.9b) \quad \text{such that } S^\top \Phi \mathbf{c} = \mathbf{y}.$$

Recalling that $\mathbf{y} = S^\top \mathbf{u}$, the solution to this constraint optimization problem is given by (see, e.g., refs. [5, 27])

$$(2.10) \quad \mathbf{c} = [\Phi^\top - \Phi^\top S(S^\top \Phi \Phi^\top S)^{-1} S^\top (\Phi \Phi^\top - \mathbb{I}_N)] \mathbf{u}.$$

It is easy to verify that $S^\top \Phi \mathbf{c} = \mathbf{y}$, and therefore, the reconstruction $\tilde{\mathbf{u}} = \Phi \mathbf{c}$ agrees with the observations $\mathbf{y} = S^\top \mathbf{u}$ at the sensor locations.

Unfortunately, expression (2.10) is still not computable since it relies on the entire state \mathbf{u} . However, in the special case where $S^\top \Phi$ is invertible, this expression simplifies to

$$(2.11) \quad \mathbf{c} = (S^\top \Phi)^{-1} S^\top \mathbf{u} = (S^\top \Phi)^{-1} \mathbf{y},$$

which is now computable from known selection matrix S , basis matrix Φ , and the observations \mathbf{y} . We note that when $S^\top \Phi$ is invertible, the constraint set (2.9b) contains the single point $\mathbf{c} = (S^\top \Phi)^{-1} \mathbf{y}$, and therefore, the solution to the optimization problem is trivial.

Expression (2.11) leads to the vanilla DEIM reconstruction,

$$(2.12) \quad \tilde{\mathbf{u}} = \Phi \mathbf{c} = \Phi (S^\top \Phi)^{-1} \mathbf{y}.$$

The reconstruction error $\|\tilde{\mathbf{u}} - \mathbf{u}\|$ can be bounded from above with a multiple of the truncation error.

LEMMA 1 (vanilla DEIM error estimate [10]). *Assume that the number of sensors and the number of modes are equal, $n = m$, and that the matrix $S^\top \Phi \in \mathbb{R}^{n \times n}$ is invertible. The vanilla DEIM reconstruction (2.12) satisfies*

$$(2.13) \quad \|\tilde{\mathbf{u}} - \mathbf{u}\| \leq \|(S^\top \Phi)^{-1}\|_2 \mathcal{E}_m(\mathbf{u}),$$

where $\mathcal{E}_m(\mathbf{u}) = \|\mathbf{u} - \hat{\mathbf{u}}\|$ is the truncation error, $\hat{\mathbf{u}} = \Phi \Phi^\top \mathbf{u}$ is the orthogonal reconstruction of \mathbf{u} , and $\|\cdot\|_2$ denotes the 2-norm (or spectral norm) of the matrix.

The truncation error \mathcal{E}_m decreases monotonically with the number of basis vectors m . Therefore, to decrease the vanilla DEIM reconstruction error, we ideally would need to use as many basis vector as possible. However, since $n = m$, the number of basis vectors m is limited by the number of available sensors.

So far, we have said nothing about the choice of the selection matrix S which determines the sensor locations. In vanilla DEIM, inequality (2.13) is used to inform the choice of the selection matrix. Ideally, the selection matrix should be chosen such that the prefactor $\|(S^\top \Phi)^{-1}\|_2$ is minimal, and therefore, the error upper bound (2.13) is as small as possible. Unfortunately, minimizing $\|(S^\top \Phi)^{-1}\|_2$ over all selection matrices is combinatorially hard, and therefore, intractable in most applications. Therefore, one instead uses a greedy algorithm to obtain a suboptimal, yet acceptable, selection matrix S . The state-of-the-art uses QR factorization with column pivoting of Φ^\top . Since sensor placement is not the focus of this paper, we refer to [14] for further details and only state the QR algorithm using MATLAB's syntax:

$$(2.14) \quad [\sim, \sim, \mathcal{I}] = \text{qr}(\Phi^\top, \text{'vector'}); \quad \mathbb{I} = \text{eye}(N); \quad S = \mathbb{I}(:, \mathcal{I}(1:n)).$$

The set $\mathcal{I}(1:n)$ contains the indices $\{i_1, \dots, i_n\}$ of the columns of the identity matrix \mathbb{I} which form the selection matrix S . The DEIM algorithm with QR factorization for sensor placement is often referred to as Q-DEIM. Here, we refer to both methods

(DEIM and Q-DEIM) as vanilla DEIM to distinguish them from the S-DEIM method introduced in section 3.

Now we turn our attention to some properties of the vanilla DEIM reconstruction (2.12). Recalling that $\mathbf{y} = S^\top \mathbf{u}$, we have $\tilde{\mathbf{u}} = \Phi(S^\top \Phi)^{-1} S^\top \mathbf{u}$. Defining the *DEIM operator* $\mathbb{D} = \Phi(S^\top \Phi)^{-1} S^\top$, we have $\tilde{\mathbf{u}} = \mathbb{D}\mathbf{u}$. We note that the DEIM operator \mathbb{D} is an oblique projection, parallel to the null space of S^\top , onto the range of Φ [15]. The following well-known result outlines some other properties of the DEIM operator.

LEMMA 2 (see [15]). *Consider the DEIM operator $\mathbb{D} = \Phi(S^\top \Phi)^{-1} S^\top$ with equal numbers of sensors and modes, $m = n$. We have the following:*

1. *Interpolation property:* $S^\top \mathbb{D} = S^\top$.
2. *Projection property:* $\mathbb{D}\Phi\Phi^\top = \Phi\Phi^\top$.

The interpolation property implies that the observations $\tilde{\mathbf{y}} = S^\top \tilde{\mathbf{u}}$ corresponding to the DEIM reconstruction agree with the true observations $\mathbf{y} = S^\top \mathbf{u}$. In other words, the interpolation property implies that the reconstruction reproduces exact values at the sensor locations, although it can incur errors elsewhere.

The projection property implies that the vanilla DEIM reconstruction of the orthogonal projection is exact, $\mathbb{D}\hat{\mathbf{u}} = \hat{\mathbf{u}}$. In particular, if $\mathbf{u} \in \mathcal{R}[\Phi]$, then its vanilla DEIM reconstruction returns the exact state since $\mathbb{D}\mathbf{u} = \mathbf{u}$.

2.2.2. Different number of sensors and modes. In this section, we comment on the application of vanilla DEIM when the number of sensors and modes are not equal, $n \neq m$. In this case, the matrix $S_n^\top \Phi_m$ is no longer square, and therefore, its inverse is not well-defined. Instead, one replaces the inverse with the Moore–Penrose pseudoinverse $(S_n^\top \Phi_m)^+$, so that the expansion coefficients are given by $\mathbf{c} = (S_n^\top \Phi_m)^+ \mathbf{y}$. Note that this coefficient vector is the *minimum-norm* solution to (2.9b). The corresponding vanilla DEIM reconstruction is given by $\tilde{\mathbf{u}} = \mathbb{D}_{m,n} \mathbf{u}$, where the DEIM operator is now defined by $\mathbb{D}_{m,n} = \Phi_m (S_n^\top \Phi_m)^+ S_n^\top$. To obtain the selection matrix S_n , one uses the same QR factorization with column pivoting as outlined in (2.14).

In practice, the matrix $S_n^\top \Phi_m$ often has full rank. In keeping with previous studies [10, 15], we make this assumption throughout this paper.

Assumption 1. For a given basis matrix Φ_m , let the selection matrix S_n be determined by the Q-DEIM algorithm (2.14). We assume that $S_n^\top \Phi_m$ has full rank, i.e., $\text{rank}[S_n^\top \Phi_m] = \min\{n, m\}$.

Remark 1 (see [15]). Unfortunately, when $n \neq m$, some of the nice properties of vanilla DEIM fall apart:

1. If $n < m$ (underdetermined, fewer sensor than modes), the interpolation property, $S_n^\top \mathbb{D}_{m,n} = S_n^\top$, still holds. However, the projection property fails, i.e., $\mathbb{D}_{m,n} \Phi_m \Phi_m^\top \neq \Phi_m \Phi_m^\top$. Furthermore, the error upper bound (2.13) is no longer valid.
2. If $n > m$ (overdetermined, more sensor than modes), the interpolation property fails, i.e., $S_n^\top \mathbb{D}_{m,n} \neq S_n^\top$. But the projection property and the error upper bound are still valid.

These statements are summarized in Table 2.

Our main focus in this paper is on the underdetermined case where the number of sensors is smaller than the number of modes, $n < m$. In this regime, under Assumption 1, the pseudoinverse $(S_n^\top \Phi_m)^+$ is, in fact, a right inverse: $(S_n^\top \Phi_m)(S_n^\top \Phi_m)^+ = \mathbb{I}_n$.

TABLE 2
Vanilla DEIM properties with various number of sensors n and modes m .

	$n = m$	$n > m$	$n < m$
Interpolation property	✓	×	✓
Projection property	✓	✓	×
Error upper bound	✓	✓	×

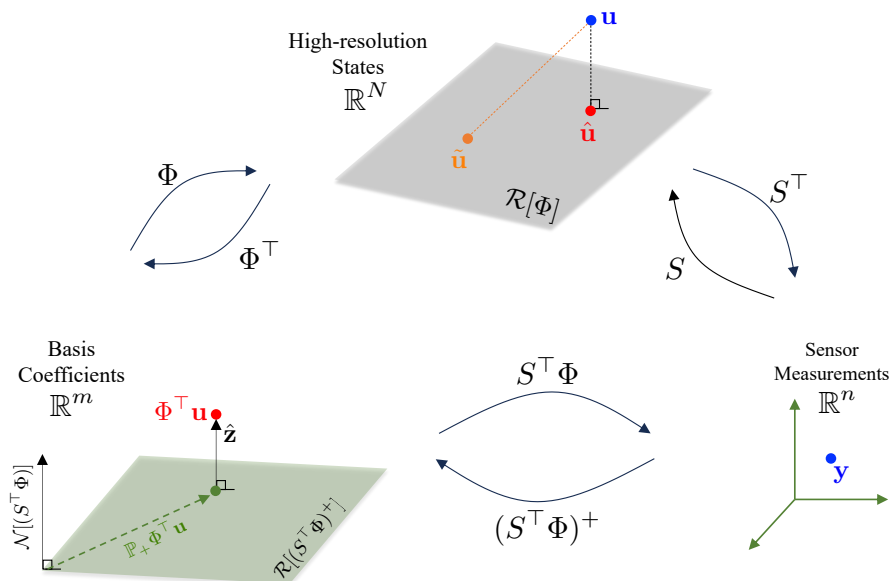


FIG. 1. Mappings between the high-resolution space \mathbb{R}^N , the measurement space \mathbb{R}^n , and the coefficient space \mathbb{R}^m . The matrix \mathbb{P}_+ is shorthand for $\mathbb{P}_{\mathcal{R}[(S^\top \Phi)^+]}$, i.e., orthogonal projection onto the range of $(S^\top \Phi)^+$. Note that the mappings in this figure do not necessarily commute.

In section 3, where we introduce sparse DEIM, we show that the appropriate error upper bound is more complicated than (2.13) and involves the null space of $S_n^\top \Phi_m$.

3. Sparse DEIM. Vanilla DEIM uses the expansion coefficients $\mathbf{c} = (S^\top \Phi)^+ \mathbf{y}$ which are the *minimum-norm* solution to the least squares problem,

$$(3.1) \quad \min_{\mathbf{c} \in \mathbb{R}^m} \|S^\top \Phi \mathbf{c} - \mathbf{y}\|^2.$$

However, the general solution to this least squares problem is given by

$$(3.2) \quad \mathbf{c}(\mathbf{z}) = (S^\top \Phi)^+ \mathbf{y} + \mathbf{z} \quad \forall \mathbf{z} \in \mathcal{N}[S^\top \Phi],$$

where \mathcal{N} denotes the null space. Note that $\mathbf{c}(\mathbf{0})$ corresponds to the vanilla DEIM reconstruction. If the number of sensors are equal to or greater than the number of modes, $n \geq m$, then we have $\mathcal{N}[S^\top \Phi] = \{\mathbf{0}\}$ and the vanilla DEIM reconstruction is the only choice. However, when the number of sensors is smaller than the number of modes, $n < m$, the null space $\mathcal{N}[S^\top \Phi]$ is an $(m - n)$ -dimensional subspace of \mathbb{R}^m and $\mathbf{z} \neq \mathbf{0}$ is a possibility. This motivates the following definition.

DEFINITION 2 (S-DEIM reconstruction). Assume there are fewer sensors than modes, $n < m$. For any nonzero $\mathbf{z} \in \mathcal{N}[S^\top \Phi]$, we refer to

$$(3.3) \quad \tilde{\mathbf{u}}(\mathbf{z}) = \Phi \mathbf{c}(\mathbf{z}) = \Phi (S^\top \Phi)^+ \mathbf{y} + \Phi \mathbf{z} = \mathbb{D} \mathbf{u} + \Phi \mathbf{z},$$

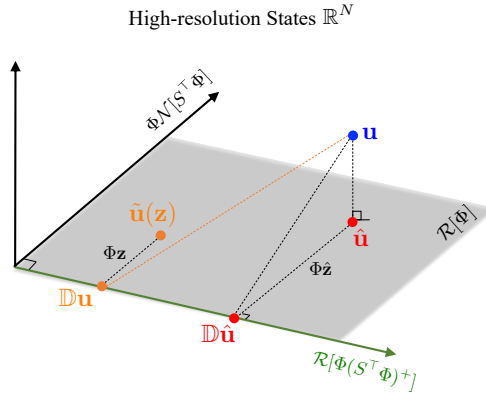


FIG. 2. Orthogonal decomposition of $\mathcal{R}[\Phi]$ with fewer sensors than modes, $n < m$. Dashed black lines represent orthogonal projections, whereas the dashed orange line represents an oblique projection. (Figure in color online.)

as the S-DEIM reconstruction of \mathbf{u} . We refer to $\mathbf{z} \in \mathcal{N}[S^\top \Phi]$ as a kernel vector.

The main question is whether there exists $\mathbf{0} \neq \mathbf{z} \in \mathcal{N}[S^\top \Phi]$ such that the corresponding S-DEIM reconstruction $\tilde{\mathbf{u}}(\mathbf{z})$ is a better approximation of \mathbf{u} than the vanilla DEIM reconstruction $\tilde{\mathbf{u}}(\mathbf{0})$?

We first note that, under Assumption 1, the addition of $\mathbf{z} \neq \mathbf{0}$ does not change the value of the reconstructed field at the sensor locations. More precisely,

$$(3.4) \quad S^\top \tilde{\mathbf{u}}(\mathbf{z}) = S^\top \Phi (S^\top \Phi)^+ \mathbf{y} + S^\top \Phi \mathbf{z} = \mathbf{y} \quad \forall \mathbf{z} \in \mathcal{N}[S^\top \Phi],$$

which implies that any S-DEIM reconstruction agrees with the data from sensor measurements, i.e., S-DEIM respects the interpolation property. As a result, the available sensor data cannot be immediately used to inform the best choice of the kernel vector \mathbf{z} . Here, we defer the choice of \mathbf{z} to sections 3.1 and 4, and first derive error estimates for the S-DEIM reconstruction with an arbitrary kernel vector $\mathbf{z} \in \mathcal{N}[S^\top \Phi]$.

Before presenting our main error estimate in Theorem 1, we need some preliminary results.

LEMMA 3. Range of $(S^\top \Phi)^+$ is the orthogonal complement of the null space of $S^\top \Phi$, i.e.,

$$(3.5) \quad \mathbb{R}^m = \mathcal{R}[(S^\top \Phi)^+] \oplus \mathcal{N}[S^\top \Phi] \quad \text{and} \quad \mathcal{R}[(S^\top \Phi)^+] \perp \mathcal{N}[S^\top \Phi]$$

Proof. This is a well-known consequence of the fundamental theorem of linear algebra (see, e.g., [35]) and the fact that $\mathcal{R}[(S^\top \Phi)^+] = \mathcal{R}[(S^\top \Phi)^\top]$. \square

As depicted in Figure 2, Lemma 3 implies that the range of Φ can be decomposed into two orthogonal subspaces as follows.

COROLLARY 1. The range of Φ admits the following orthogonal decomposition:

$$(3.6) \quad \mathcal{R}[\Phi] = \mathcal{R}[\Phi(S^\top \Phi)^+] \oplus \Phi \mathcal{N}[S^\top \Phi] \quad \text{and} \quad \mathcal{R}[\Phi(S^\top \Phi)^+] \perp \Phi \mathcal{N}[S^\top \Phi],$$

where $\Phi \mathcal{N}[S^\top \Phi] = \{\mathbf{u} \in \mathbb{R}^N : \mathbf{u} = \Phi \mathbf{z}, \mathbf{z} \in \mathcal{N}[S^\top \Phi]\}$.

Proof. This follows from the fact that the columns of Φ are orthonormal. In particular, it is injective and it preserves the Euclidean inner product ($\Phi^\top \Phi = \mathbb{I}_m$). \square

DEFINITION 3 (kernel matrix). Let $\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_{m-n}\} \subset \mathbb{R}^m$ be an orthonormal basis for $\mathcal{N}[S^\top \Phi]$. We refer to the matrix $Z = [\mathbf{z}_1 | \mathbf{z}_2 | \dots | \mathbf{z}_{m-n}]$ as a kernel matrix.

The following lemma gives the best choice of the kernel vector $\mathbf{z} \in \mathcal{N}[S^\top \Phi]$ in terms of the kernel matrix Z .

LEMMA 4. The optimization problem,

$$(3.7) \quad \hat{\mathbf{z}} = \underset{\mathbf{z} \in \mathcal{N}[S^\top \Phi]}{\operatorname{argmin}} \|\tilde{\mathbf{u}}(\mathbf{z}) - \mathbf{u}\|^2,$$

has a unique solution given by $\hat{\mathbf{z}} = ZZ^\top \Phi^\top \mathbf{u}$, where Z is a kernel matrix.

Proof. Consider a kernel matrix Z and note that, for any $\mathbf{z} \in \mathcal{N}[S^\top \Phi]$, there exists a unique $\boldsymbol{\xi} \in \mathbb{R}^{m-n}$ such that $\mathbf{z} = Z\boldsymbol{\xi}$. Therefore, optimization problem (3.7) can be equivalently written as

$$(3.8) \quad \min_{\boldsymbol{\xi} \in \mathbb{R}^{m-n}} \|\Phi(S^\top \Phi)^+ \mathbf{y} + \Phi Z \boldsymbol{\xi} - \mathbf{u}\|^2.$$

Since ΦZ has full column rank, the solution is unique. Furthermore, since $(\Phi Z)^+ = Z^\top \Phi^\top$, the solution is given by

$$(3.9) \quad \hat{\boldsymbol{\xi}} = Z^\top \Phi^\top \mathbf{u} - Z^\top \Phi^\top \Phi (S^\top \Phi)^+ \mathbf{y}.$$

Next, we use the fact that $\Phi^\top \Phi = \mathbb{I}_m$ and $Z^\top (S^\top \Phi)^+ = 0$. The latter identity follows from Lemma 3: $\mathcal{R}(Z) = \mathcal{N}[S^\top \Phi] \perp \mathcal{R}[(S^\top \Phi)^+]$. Therefore, we have $\hat{\boldsymbol{\xi}} = Z^\top \Phi^\top \mathbf{u}$ and $\hat{\mathbf{z}} = ZZ^\top \Phi^\top \mathbf{u}$. Note that, although the kernel matrix Z is not necessarily unique, ZZ^\top is the orthogonal projection onto $\mathcal{N}[S^\top \Phi]$, and therefore, $\hat{\mathbf{z}}$ is unique. \square

Lemma 4 gives the best choice of the kernel vector \mathbf{z} which minimizes the error of the S-DEIM reconstruction $\tilde{\mathbf{u}}(\mathbf{z})$. Unfortunately, the optimal kernel vector $\hat{\mathbf{z}} = ZZ^\top \Phi^\top \mathbf{u}$ is not computable from observations since it requires the entire state \mathbf{u} . In section 4, we propose a computable method for estimating the optimal kernel vector. But first, we state our main error estimates of the S-DEIM reconstruction for arbitrary kernel vectors.

THEOREM 1 (S-DEIM error). Let the number of sensors be smaller than the number of modes, $n < m$. For $\mathbf{u} \in \mathbb{R}^N$, let $\hat{\mathbf{u}} = \Phi \Phi^\top \mathbf{u}$ denote its orthogonal projection onto $\mathcal{R}[\Phi]$. Let $\tilde{\mathbf{u}}(\mathbf{z})$ denote the S-DEIM reconstruction for an arbitrary kernel vector $\mathbf{z} \in \mathcal{N}[S^\top \Phi]$, and $\hat{\mathbf{z}} = ZZ^\top \Phi^\top \mathbf{u}$ as in Lemma 4. Then

$$(3.10) \quad \|\mathbf{u} - \tilde{\mathbf{u}}(\mathbf{z})\|^2 = \|\mathbf{u} - \hat{\mathbf{u}}\|^2 + \|\mathbb{D}(\mathbf{u} - \hat{\mathbf{u}})\|^2 + \|\hat{\mathbf{z}} - \mathbf{z}\|^2 \quad \forall \mathbf{z} \in \mathcal{N}[S^\top \Phi].$$

Proof. First, we write $\mathbf{u} - \tilde{\mathbf{u}}(\mathbf{z}) = \mathbf{u} - \hat{\mathbf{u}} + \hat{\mathbf{u}} - \tilde{\mathbf{u}}(\mathbf{z})$ where $\hat{\mathbf{u}} = \Phi \Phi^\top \mathbf{u}$ and $\tilde{\mathbf{u}}(\mathbf{z}) = \Phi \mathbf{c}(\mathbf{z})$ both belong to $\mathcal{R}[\Phi]$; therefore, $\hat{\mathbf{u}} - \tilde{\mathbf{u}}(\mathbf{z}) \in \mathcal{R}[\Phi]$. On the other hand, $\mathbf{u} - \hat{\mathbf{u}} = (\mathbb{I} - \Phi \Phi^\top) \mathbf{u}$ is orthogonal to $\mathcal{R}[\Phi]$. Therefore,

$$(3.11) \quad \|\mathbf{u} - \tilde{\mathbf{u}}(\mathbf{z})\|^2 = \|\mathbf{u} - \hat{\mathbf{u}} + \hat{\mathbf{u}} - \tilde{\mathbf{u}}(\mathbf{z})\|^2 = \|\mathbf{u} - \hat{\mathbf{u}}\|^2 + \|\hat{\mathbf{u}} - \tilde{\mathbf{u}}(\mathbf{z})\|^2.$$

Next, we focus on $\hat{\mathbf{u}} - \tilde{\mathbf{u}}(\mathbf{z})$. Recall from Lemma 3 that $\mathbb{R}^m = \mathcal{N}[S^\top \Phi] \oplus \mathcal{R}[(S^\top \Phi)^+]$ and that the two subspaces are orthogonal. Write $\Phi^\top \mathbf{u} \in \mathbb{R}^m$ as its unique orthogonal decomposition,

$$(3.12) \quad \Phi^\top \mathbf{u} = \mathbb{P}_{\mathcal{R}[(S^\top \Phi)^+]} \Phi^\top \mathbf{u} + \hat{\mathbf{z}},$$

where $\hat{\mathbf{z}} \in \mathcal{N}[S^\top \Phi]$ is unique. Note that the orthogonal projection $\mathbb{P}_{\mathcal{R}[(S^\top \Phi)^+]}$ onto $\mathcal{R}[(S^\top \Phi)^+]$ can be written as

$$(3.13) \quad \mathbb{P}_{\mathcal{R}[(S^\top \Phi)^+]} = (S^\top \Phi)^+ (S^\top \Phi),$$

which implies

$$(3.14) \quad \hat{\mathbf{u}} = \Phi \Phi^\top \mathbf{u} = \Phi (S^\top \Phi)^+ (S^\top \Phi) \Phi^\top \mathbf{u} + \Phi \hat{\mathbf{z}} = \mathbb{D} \Phi \Phi^\top \mathbf{u} + \Phi \hat{\mathbf{z}} = \mathbb{D} \hat{\mathbf{u}} + \Phi \hat{\mathbf{z}}.$$

As a side note, we mention that, if $n = m$ and $S^\top \Phi$ is invertible, then $\mathbb{D} \Phi \Phi^\top = \Phi \Phi^\top$. However, in our case where $n < m$, the identity does not hold (cf. Remark 1).

Returning to the proof and recalling that $\tilde{\mathbf{u}}(\mathbf{z}) = \Phi \mathbf{c}(\mathbf{z}) = \mathbb{D} \mathbf{u} + \Phi \mathbf{z}$, we have

$$(3.15) \quad \begin{aligned} \|\hat{\mathbf{u}} - \tilde{\mathbf{u}}(\mathbf{z})\|^2 &= \|\mathbb{D} \hat{\mathbf{u}} - \mathbb{D} \mathbf{u} + \Phi \hat{\mathbf{z}} - \Phi \mathbf{z}\|^2 \\ &= \|\mathbb{D} \hat{\mathbf{u}} - \mathbb{D} \mathbf{u}\|^2 + \|\Phi \hat{\mathbf{z}} - \Phi \mathbf{z}\|^2 \\ &= \|\mathbb{D} \hat{\mathbf{u}} - \mathbb{D} \mathbf{u}\|^2 + \|\hat{\mathbf{z}} - \mathbf{z}\|^2, \end{aligned}$$

where we used the fact that $\Phi \hat{\mathbf{z}} - \Phi \mathbf{z} \in \Phi \mathcal{N}[S^\top \Phi]$ and $\mathbb{D} \Phi \Phi^\top \mathbf{u} - \mathbb{D} \mathbf{u} \in \mathcal{R}[\Phi (S^\top \Phi)^+]$ are orthogonal (cf. Corollary 1).

It remains to show that $\hat{\mathbf{z}} = Z Z^\top \Phi^\top \mathbf{u}$. Recall from (3.12) and (3.13) that

$$(3.16) \quad \Phi^\top \mathbf{u} = (S^\top \Phi)^+ S^\top \Phi \Phi^\top \mathbf{u} + \hat{\mathbf{z}}.$$

For any $\hat{\mathbf{z}} \in \mathcal{N}[S^\top \Phi]$ there exists a unique $\boldsymbol{\xi} \in \mathbb{R}^{n-m}$ such that $\hat{\mathbf{z}} = Z \boldsymbol{\xi}$. Substituting this in the above equation, and noting that $\mathcal{N}[S^\top \Phi]$ is orthogonal to $\mathcal{R}[(S^\top \Phi)^+]$, we obtain $\boldsymbol{\xi} = Z^\top \Phi^\top \mathbf{u}$. \square

Theorem 1 has a number of important consequences. For instance, if the true state \mathbf{u} belongs to the range of Φ , then there exists an exact S-DEIM reconstruction.

COROLLARY 2. *Let the number of sensors be smaller than the number of modes, $n < m$. If $\mathbf{u} \in \mathcal{R}(\Phi)$, then the S-DEIM reconstruction with $\mathbf{z} = \hat{\mathbf{z}} = Z Z^\top \Phi^\top \mathbf{u}$ is exact: $\tilde{\mathbf{u}}(\hat{\mathbf{z}}) = \mathbf{u}$.*

Proof. Choose $\mathbf{z} = \hat{\mathbf{z}}$ in Theorem 1. Since $\mathbf{u} \in \mathcal{R}(\Phi)$, we have $\hat{\mathbf{u}} = \mathbf{u}$, and therefore, $\|\mathbf{u} - \tilde{\mathbf{u}}(\hat{\mathbf{z}})\| = 0$. \square

Remark 2. Corollary 2 shows that, if $\mathbf{u} \in \mathcal{R}(\Phi)$, then the S-DEIM reconstruction is exact. Note, however, that when $n < m$ the vanilla DEIM reconstruction $\mathbb{D} \mathbf{u}$ is not exact, $\mathbb{D} \mathbf{u} \neq \mathbf{u}$, even if $\mathbf{u} \in \mathcal{R}(\Phi)$. This is due to the fact that the vanilla DEIM reconstruction fails to preserve the projection property as discussed in Remark 1, i.e., $\mathbb{D} \Phi \Phi^\top \neq \Phi \Phi^\top$.

Although \mathbb{D} is not an orthogonal projection, it is straightforward to verify that $\mathbb{D} \Phi \Phi^\top = \Phi (S^\top \Phi)^+ (S^\top \Phi) \Phi^\top$ is an orthogonal projection onto $\mathcal{R}[\Phi (S^\top \Phi)^+]$. As a result, even if $\hat{\mathbf{u}} \neq \mathbf{u}$, $\mathbb{D} \hat{\mathbf{u}} = \mathbb{D} \Phi \Phi^\top \mathbf{u}$ is the orthogonal projection of \mathbf{u} onto $\mathcal{R}[\Phi (S^\top \Phi)^+]$. Figure 2 shows this schematically.

Theorem 1 also leads to an upper bound for the S-DEIM reconstruction error. This upper bound is a generalization of (2.13) to the case where the number of sensors is smaller than the number of modes.

COROLLARY 3 (error estimate for S-DEIM). *Let the number of sensors be smaller than the number of modes, $n < m$. The S-DEIM reconstruction satisfies*

$$(3.17) \quad \|\tilde{\mathbf{u}}(\mathbf{z}) - \mathbf{u}\| \leq \|(S^\top \Phi)^+\|_2 \mathcal{E}_m(\mathbf{u}) + \|\mathbf{z} - \hat{\mathbf{z}}\| \quad \forall \mathbf{z} \in \mathcal{N}[S^\top \Phi],$$

where $\mathcal{E}_m(\mathbf{u}) = \|\hat{\mathbf{u}} - \mathbf{u}\|$ is the truncation error and $\hat{\mathbf{z}} = Z Z^\top \Phi^\top \mathbf{u}$ is the optimal kernel vector.

Proof. We recall from the proof of Theorem 1 that $\tilde{\mathbf{u}}(\mathbf{z}) = \mathbb{D}\mathbf{u} + \Phi\mathbf{z}$ and $\hat{\mathbf{u}} = \mathbb{D}\hat{\mathbf{u}} + \Phi\hat{\mathbf{z}}$. Therefore, we have $\tilde{\mathbf{u}}(\mathbf{z}) - \mathbf{u} = (\mathbb{I} - \mathbb{D})(\hat{\mathbf{u}} - \mathbf{u}) + \Phi(\mathbf{z} - \hat{\mathbf{z}})$. Using the triangle inequality, we obtain

$$\begin{aligned} \|\tilde{\mathbf{u}}(\mathbf{z}) - \mathbf{u}\| &\leq \|(\mathbb{I} - \mathbb{D})(\hat{\mathbf{u}} - \mathbf{u})\| + \|\Phi(\mathbf{z} - \hat{\mathbf{z}})\| \\ (3.18) \quad &\leq \|\mathbb{D}\|_2 \mathcal{E}_m(\mathbf{u}) + \|\mathbf{z} - \hat{\mathbf{z}}\|, \end{aligned}$$

where in the last inequality, we used the fact that \mathbb{D} is a nonzero and nonidentity projection, $\mathbb{D}^2 = \mathbb{D}$. Any such projection satisfies $\|\mathbb{D}\|_2 = \|\mathbb{I} - \mathbb{D}\|_2$ [21, 36]. Finally, we use the fact that the DEIM operator $\mathbb{D} = \Phi(S^\top \Phi)^+ S^\top$ satisfies $\|\mathbb{D}\|_2 \leq \|(S^\top \Phi)^+\|_2$ since $\|\Phi\|_2 = \|S^\top\|_2 = 1$. \square

When the number of sensors is equal to the number of modes, $n = m$, we have $\mathcal{N}[S^\top \Phi] = \{\mathbf{0}\}$, and therefore, $\mathbf{z} = \hat{\mathbf{z}} = \mathbf{0}$. As a result, the last term on the right-hand side of (3.17) vanishes. In this special case, the error upper bound coincides with the one in Lemma 1 for vanilla DEIM. However, when $n < m$, the correct upper bound is the one given in (3.17) involving the difference between the kernel vector \mathbf{z} and its optimal value $\hat{\mathbf{z}}$.

Remark 3. The error upper bound (3.17) highlights an important advantage of S-DEIM over underdetermined vanilla DEIM ($n < m$). Assume that a kernel vector \mathbf{z} is chosen such that the kernel error is small, $\|\mathbf{z} - \hat{\mathbf{z}}\| < \epsilon$. By increasing the number of modes m , the S-DEIM reconstruction error will eventually become smaller than ϵ . This is because the truncation error $\mathcal{E}_m(\mathbf{u})$ is a monotonically decreasing function of m . Furthermore, for a fixed number of sensors, the spectral norm $\|(S_n^\top \Phi_m)^+\|_2$ decreases as m increases (This follows from interlacing inequalities for singular values; see, e.g., Theorem 7.3.9 of [19]). Such a small error cannot be guaranteed for underdetermined vanilla DEIM since $\mathbf{z} = \mathbf{0}$, and therefore, the upper bound is always larger than $\|\hat{\mathbf{z}}\|$.

In this section, we largely sidestepped the choice of the kernel vector \mathbf{z} . In section 4, we propose a computable method for obtaining an optimal kernel vector \mathbf{z} which minimizes the S-DEIM reconstruction error. But first, in section 3.1 below, we discuss a promising idea for selecting the kernel vector and show that it unfortunately fails to lead to any improvement over vanilla DEIM.

3.1. Withholding measurements would not work. Assume that n distinct sensor measurements are available. We split these measurements into two batches with n_1 and n_2 measurement in each so that $n_1 + n_2 = n$. We withhold the n_2 measurements at first and use n_1 sensor measurements to obtain the S-DEIM reconstruction $\tilde{\mathbf{u}}(\mathbf{z})$ with an unknown kernel vector \mathbf{z} . Then the remaining n_2 withheld measurements are used to compute the kernel vector \mathbf{z} .

To describe the method more precisely, consider a selection matrix $S \in \mathbb{R}^{N \times n}$ which identifies the position of n distinct sensors. This selection matrix, for instance, can be obtained by the column pivoted QR factorization (2.14). We split the selection matrix into two blocks, $S = [S_1 \ S_2]$, where $S_1 \in \mathbb{R}^{N \times n_1}$ corresponds to the first batch of n_1 sensors and $S_2 \in \mathbb{R}^{N \times n_2}$ corresponds to the second batch. Note that we use the shorthand notation S_i for S_{n_i} . Furthermore, we assume, without loss of generality, that the first batch corresponds to the first n_1 columns of S . We write the measurements \mathbf{y} as

$$(3.19) \quad \mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix},$$

where $\mathbf{y}_1 \in \mathbb{R}^{n_1}$ denotes the measurements from the first batch of sensors and $\mathbf{y}_2 \in \mathbb{R}^{n_2}$ denotes the remaining measurements. Note that $\mathbf{y}_i = S_i^\top \mathbf{u}$, $i = 1, 2$, where \mathbf{u} is the unknown true state.

With this set-up, and only using the first n_1 sensors, the S-DEIM coefficients are given by

$$(3.20) \quad \mathbf{c}(\mathbf{z}) = (S_1^\top \Phi)^+ \mathbf{y}_1 + \mathbf{z},$$

where $\mathbf{z} \in \mathcal{N}[S_1^\top \Phi]$ is the kernel vector to be determined. The reconstructed state is given by $\tilde{\mathbf{u}}(\mathbf{z}) = \Phi \mathbf{c}(\mathbf{z})$. In order to find the kernel vector \mathbf{z} , we use the second batch of sensors. More precisely, we determine \mathbf{z} by requiring $S_2^\top \tilde{\mathbf{u}}(\mathbf{z})$, i.e., the reconstructed state measured at the second batch of sensors, to be as close to the true measurements \mathbf{y}_2 as possible. In other words, we solve the optimization problem,

$$(3.21) \quad \overline{\min}_{\mathbf{z} \in \mathcal{N}[S_1^\top \Phi]} \|S_2^\top \Phi \mathbf{c}(\mathbf{z}) - \mathbf{y}_2\|^2,$$

where the overline indicates the minimum-norm solution. As the following theorem states, the solution to (3.21) coincides with the solution to vanilla DEIM reconstruction (2.12) using all sensors at once.

THEOREM 2 (equivalence of vanilla DEIM and two-stage S-DEIM). *Consider an S-DEIM reconstruction using n_1 sensors, $\tilde{\mathbf{u}}(\mathbf{z}) = \Phi \mathbf{c}(\mathbf{z})$ where $\mathbf{c}(\mathbf{z}) = (S_1^\top \Phi)^+ \mathbf{y}_1 + \mathbf{z}$. Determine the kernel vector $\mathbf{z} \in \mathcal{N}[S_1^\top \Phi]$ as the solution of (3.21). Under Assumption 1, $\tilde{\mathbf{u}}(\mathbf{z})$ coincides with the vanilla DEIM reconstruction $\Phi(S^\top \Phi)^+ \mathbf{y}$.*

Proof. We first write the minimization problem (3.21) more explicitly. Let $Z_1 \in \mathbb{R}^{m \times (m-n_1)}$ denote a matrix whose columns form an orthonormal basis for $\mathcal{N}[S_1^\top \Phi]$. Any $\mathbf{z} \in \mathcal{N}[S_1^\top \Phi]$ can be written uniquely as $\mathbf{z} = Z_1 \boldsymbol{\xi}$ for some $\boldsymbol{\xi} \in \mathbb{R}^{m-n_1}$. Therefore, optimization problem (3.21) can be written equivalently as

$$(3.22) \quad \overline{\min}_{\boldsymbol{\xi} \in \mathbb{R}^{m-n_1}} \|S_2^\top \Phi (S_1^\top \Phi)^+ \mathbf{y}_1 + S_2^\top \Phi Z_1 \boldsymbol{\xi} - \mathbf{y}_2\|^2$$

Now we turn our attention to vanilla DEIM. Recall that vanilla DEIM corresponds to the minimum-norm minimizer of the cost function

$$(3.23) \quad J(\mathbf{c}) = \|S^\top \Phi \mathbf{c} - \mathbf{y}\|^2.$$

First, we write this cost function in a slightly different but equivalent form. Note that, for all $\mathbf{c} \in \mathbb{R}^m$, we have

$$(3.24) \quad S^\top \Phi \mathbf{c} = \begin{bmatrix} S_1^\top \Phi \mathbf{c} \\ S_2^\top \Phi \mathbf{c} \end{bmatrix},$$

and therefore,

$$(3.25) \quad J(\mathbf{c}) = \|S_1^\top \Phi \mathbf{c} - \mathbf{y}_1\|^2 + \|S_2^\top \Phi \mathbf{c} - \mathbf{y}_2\|^2.$$

Next, we decompose \mathbf{c} into its orthogonal projections onto the subspaces $\mathcal{R}[(S_1^\top \Phi)^+]$ and $\mathcal{N}[S_1^\top \Phi]$. Recall that these are subspaces of \mathbb{R}^m and are orthogonal complements of each other. Therefore, for any $\mathbf{c} \in \mathbb{R}^m$, there exist $\boldsymbol{\eta} \in \mathbb{R}^{n_1}$ and $\boldsymbol{\xi} \in \mathbb{R}^{m-n_1}$ such that $\mathbf{c} = (S_1^\top \Phi)^+ \boldsymbol{\eta} + Z_1 \boldsymbol{\xi}$. Recalling that $S_1^\top \Phi (S_1^\top \Phi)^+ = \mathbb{I}_{n_1}$, we have

$$(3.26a) \quad S_1^\top \Phi \mathbf{c} = \boldsymbol{\eta},$$

$$(3.26b) \quad S_2^\top \Phi \mathbf{c} = S_2^\top \Phi (S_1^\top \Phi)^+ \boldsymbol{\eta} + S_2^\top \Phi Z_1 \boldsymbol{\xi}.$$

Substituting these in (3.25), we obtain

$$(3.27) \quad J(\mathbf{c}) = J(\boldsymbol{\eta}, \boldsymbol{\xi}) = \|\boldsymbol{\eta} - \mathbf{y}_1\|^2 + \|S_2^\top \Phi(S_1^\top \Phi)^+ \boldsymbol{\eta} + S_2^\top \Phi Z_1 \boldsymbol{\xi} - \mathbf{y}_2\|^2.$$

Therefore, the vanilla DEIM optimization $\min_{\mathbf{c} \in \mathbb{R}^m} J(\mathbf{c})$ can be written equivalently as

$$(3.28) \quad \min_{\boldsymbol{\eta} \in \mathbb{R}^{n_1}, \boldsymbol{\xi} \in \mathbb{R}^{m-n_1}} J(\boldsymbol{\eta}, \boldsymbol{\xi}).$$

We denote the minimizer by $(\boldsymbol{\eta}_*, \boldsymbol{\xi}_*)$. Now let $\mathbf{c}_* = (S^\top \Phi)^+ \mathbf{y}$ denote the vanilla DEIM minimizer of $J(\mathbf{c})$. Recall that the vanilla DEIM cost function attains $J(\mathbf{c}_*) = 0$ at its minimum. Since $J(\mathbf{c}) = J(\boldsymbol{\eta}, \boldsymbol{\xi})$, we must also have $J(\boldsymbol{\eta}_*, \boldsymbol{\xi}_*) = 0$. Examining (3.27), for $J(\boldsymbol{\eta}_*, \boldsymbol{\xi}_*) = 0$ to be achieved, it is necessary that $\boldsymbol{\eta}_* = \mathbf{y}_1$. Therefore, optimization problem (3.28) is equivalent to

$$(3.29) \quad \min_{\boldsymbol{\xi} \in \mathbb{R}^{m-n_1}} \|S_2^\top \Phi(S_1^\top \Phi)^+ \mathbf{y}_1 + S_2^\top \Phi Z_1 \boldsymbol{\xi} - \mathbf{y}_2\|^2,$$

where we substituted $\boldsymbol{\eta} = \mathbf{y}_1$. This optimization problem is equivalent to vanilla DEIM and identical to the optimization problem (3.22). Therefore, the two-stage S-DEIM is equivalent to the vanilla DEIM reconstruction. \square

4. Data assimilation with S-DEIM. So far we have not specified how the states \mathbf{u} are generated. In this section, we assume that the states are generated by a continuous-time dynamical system. In this case, we propose a method for estimating the unknown kernel vector of S-DEIM using the observational time series $\mathbf{y}(t)$. The resulting method, which we refer to as the *Data Assimilated Sparse DEIM* (DAS-DEIM), amounts to a fast data assimilation method for dynamical systems.

Consider a system of ODEs, possibly arising from the spatial discretization of a partial differential equation,

$$(4.1) \quad \frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}), \quad \mathbf{u}(0) = \mathbf{u}_0,$$

where $\mathbf{f} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is a known Lipschitz continuous vector field. We denote the solution of the system by $\mathbf{u}(t; \mathbf{u}_0) = \varphi^t(\mathbf{u}_0)$ where $\varphi^t : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is the flow map associated with (4.1). For notational simplicity, we write $\mathbf{u}(t)$ and omit its dependence on the initial condition \mathbf{u}_0 . The corresponding observational time series are given by $\mathbf{y}(t) = S^\top \mathbf{u}(t)$. If the dynamical system is dissipative, its trajectories converge to an attractor $\mathcal{A} \subset \mathbb{R}^N$ whose dimension is smaller than N . If such a lower-dimensional attractor does not exist, we set $\mathcal{A} = \mathbb{R}^N$.

In our setting, the initial condition \mathbf{u}_0 is not completely known. Instead, only its entries corresponding to the sensor locations are known which yield the observations $\mathbf{y}(0) = S^\top \mathbf{u}_0$ at the initial time. As a result, ODE (4.1) cannot be directly integrated to estimate the future states $\mathbf{u}(t)$.

Nonetheless, at any time instance t , S-DEIM can be used to compute the state estimation,

$$(4.2) \quad \tilde{\mathbf{u}}(\mathbf{z}(t)) = \Phi \mathbf{c}(\mathbf{z}(t)) = \Phi (S^\top \Phi)^+ \mathbf{y}(t) + \Phi \mathbf{z}(t),$$

where $\mathbf{z}(t) \in \mathcal{N}[S^\top \Phi]$ for all times $t \geq 0$. The kernel vector $\mathbf{z}(t)$ is still unknown and our task is to determine its optimal value. To this end, standard data assimilation methods, such as three- and four-dimensional variational data assimilation (3DVAR and 4DVAR, respectively), can be used (cf. [4] for a review).

However, these data assimilation methods rely on descent iterations to minimize a suitable cost function. At each iteration, the descent direction needs to be approximated which requires repeated solves of the ODE (4.1). In the case of 4DVAR, an additional adjoint ODE also needs to be solved at each iteration [4]. Since the dimension N of the system is often large, this leads to a high computational cost. Furthermore, the cost function is a nonconvex function of the kernel vector \mathbf{z} ; therefore, the optimization may never converge to the global minimizer. The high computational cost and nonconvexity of the existing data assimilation methods hinders their applicability to high-dimensional systems. Here, we introduce an alternative approach which circumvents these two issues. Our method seeks $\mathbf{z}(t)$ such that the corresponding S-DEIM reconstruction $\tilde{\mathbf{u}}(\mathbf{z}(t))$ solves the ODE (4.1) as closely as possible.

More precisely, consider a path $\mathbf{z}(t) \in \mathcal{N}[S^\top \Phi]$ and its corresponding S-DEIM reconstruction (4.2). Since S-DEIM is an approximation, $\tilde{\mathbf{u}}(\mathbf{z}(t))$ does not necessarily solve the ODE (4.1). Instead, we seek $\mathbf{z}(t)$ such that the error between the left- and right-hand side of the ODE is instantaneously minimized. In particular, we solve the optimization problem,

$$(4.3) \quad \min_{\mathbf{z}(t)} \left\| \frac{d}{dt} \tilde{\mathbf{u}}(\mathbf{z}(t)) - \mathbf{f}(\tilde{\mathbf{u}}(\mathbf{z}(t))) \right\|^2,$$

where

$$(4.4) \quad \frac{d}{dt} \tilde{\mathbf{u}}(\mathbf{z}(t)) = \Phi (S^\top \Phi)^+ \dot{\mathbf{y}}(t) + \Phi \dot{\mathbf{z}}(t).$$

At first glance, this approach may seem problematic since the time derivative of the S-DEIM reconstruction involves the time derivative of the observations $\dot{\mathbf{y}}$. As such, any noise in the observations can be detrimental. However, the next theorem shows that, not only the solution to the optimization problem (4.3) can be written explicitly, but also that the solution is independent of the derivative of the observations.

THEOREM 3. *Optimization problem (4.3) has a unique solution given by $\mathbf{z}(t) = Z\boldsymbol{\xi}(t)$ where $\boldsymbol{\xi}(t) \in \mathbb{R}^{m-n}$ solves the ODE,*

$$(4.5) \quad \dot{\boldsymbol{\xi}} = Z^\top \Phi^\top \mathbf{f}(\tilde{\mathbf{u}}(\boldsymbol{\xi})),$$

where Z is a kernel matrix and $\tilde{\mathbf{u}}(\boldsymbol{\xi})$ is shorthand notation for $\tilde{\mathbf{u}}(Z\boldsymbol{\xi})$.

Proof. Recall that the columns of Z are orthonormal and $\mathcal{R}[Z] = \mathcal{N}[S^\top \Phi]$. Consequently, for any $\mathbf{z}(t) \in \mathcal{N}[S^\top \Phi]$, there exist a unique $\boldsymbol{\xi}(t) \in \mathbb{R}^{m-n}$ such that $\mathbf{z}(t) = Z\boldsymbol{\xi}(t)$. Therefore, using (4.4), optimization problem (4.3) can be written equivalently as

$$(4.6) \quad \min_{\boldsymbol{\xi}(t) \in \mathbb{R}^{m-n}} \left\| \Phi (S^\top \Phi)^+ \dot{\mathbf{y}}(t) + \Phi Z \dot{\boldsymbol{\xi}}(t) - \mathbf{f}(\tilde{\mathbf{u}}(\boldsymbol{\xi}(t))) \right\|^2.$$

Since ΦZ has full column rank and $(\Phi Z)^+ = Z^\top \Phi^\top$, the minimizer is unique and is given by

$$(4.7) \quad \dot{\boldsymbol{\xi}} = -Z^\top \Phi^\top \Phi (S^\top \Phi)^+ \dot{\mathbf{y}}(t) + Z^\top \Phi^\top \mathbf{f}(\tilde{\mathbf{u}}(\boldsymbol{\xi})).$$

Finally, since $\Phi^\top \Phi = \mathbb{I}_m$ and $Z^\top (S^\top \Phi)^+ = 0$ (due to Lemma 3), we obtain the desired result. \square

Algorithm 4.1. Data assimilated sparse DEIM (DAS-DEIM) algorithm.

Inputs:

- Basis matrix $\Phi = [\phi_1 | \phi_2 | \dots | \phi_m] \in \mathbb{R}^{N \times m}$
- Observations $\mathbf{y}(t) \in \mathbb{R}^n$, $0 \leq t \leq T$
- Vector field $\mathbf{f} : \mathbb{R}^N \rightarrow \mathbb{R}^N$

1. $\mathbf{S} = \text{pivotQR}(\Phi)$ ▷ Compute the selection matrix using (2.14)
2. $\mathbf{Z} = \text{null}(\mathbf{S}^\top \Phi)$ ▷ Compute the kernel matrix
3. Choose an initial guess: $\boldsymbol{\xi}_0 \in \mathbb{R}^{m-n}$
4. Solve $\dot{\boldsymbol{\xi}} = \mathbf{Z}^\top \Phi^\top \mathbf{f}(\tilde{\mathbf{u}}(\boldsymbol{\xi}))$ with $\boldsymbol{\xi}(0) = \boldsymbol{\xi}_0$. ▷ Solve for $\boldsymbol{\xi}(t)$, $0 \leq t \leq T$
5. State reconstruction: $\tilde{\mathbf{u}}(t) = \Phi (\mathbf{S}^\top \Phi)^\dagger \mathbf{y}(t) + \Phi \mathbf{Z} \boldsymbol{\xi}(t)$, $0 \leq t \leq T$

Outputs: DAS-DEIM reconstruction $\tilde{\mathbf{u}}(t)$, $0 \leq t \leq T$

We refer to (4.5) as the *kernel ODE* and the corresponding approximation (4.2) as the DAS-DEIM reconstruction. Since the solution (4.5) minimizes the instantaneous error between the left- and right-hand side of the governing ODE (4.1), one expects that, over time, the corresponding DAS-DEIM reconstruction $\tilde{\mathbf{u}}(\boldsymbol{\xi}(t))$ evolves closer to the true state $\mathbf{u}(t)$. More precisely, consider the observations $\{\mathbf{y}(t), 0 \leq t \leq T\}$ from the initial time to the current time T . Using this observations in its right-hand side, the ODE (4.5) can be solved numerically from an initial guess $\boldsymbol{\xi}(0) = \boldsymbol{\xi}_0 \in \mathbb{R}^{m-n}$ for the kernel vector. Even if the initial guess $\boldsymbol{\xi}_0$ is not suitable, we expect that over time it converges towards its optimal value so that the reconstruction error $\|\tilde{\mathbf{u}}(\boldsymbol{\xi}(t)) - \mathbf{u}(t)\|$ decreases as time t increases. This procedure is summarized in Algorithm 4.1.

The dimension of the kernel ODE (4.5) is $m - n$, which is much smaller than the dimension N of the original ODE (4.1). While standard data assimilation methods, such as 3DVAR and 4DVAR, require repeated solves of the N -dimensional ODE (4.1), DAS-DEIM requires solving the smaller kernel ODE only once. As a result, DAS-DEIM is computationally more efficient than standard data assimilation techniques. Nonetheless, numerically integrating the kernel ODE (4.5) still requires the evaluation of the N -dimensional vector field \mathbf{f} .

We show in section 5, with two numerical examples, that the reconstruction error of DAS-DEIM does indeed decrease over time. But before presenting our numerical results, we prove that, under certain conditions, the reconstruction error of DAS-DEIM converges exponentially fast towards zero as $t \rightarrow \infty$.

4.1. Convergence of DAS-DEIM. In order to state our main convergence result, we recall the notion of one-sided Lipschitz continuity.

DEFINITION 4. A vector field $\mathbf{f} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is one-sided Lipschitz continuous if there exists $\rho_f \in \mathbb{R}$ such that

$$(4.8) \quad \langle \tilde{\mathbf{u}} - \mathbf{u}, \mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u}) \rangle \leq \rho_f \|\tilde{\mathbf{u}} - \mathbf{u}\|^2$$

for all $\tilde{\mathbf{u}}, \mathbf{u} \in \mathbb{R}^N$. The coefficient ρ_f is referred to as the one-sided Lipschitz constant.

Remark 4. A few remarks about Definition 4 are in order:

1. Note that $\tilde{\mathbf{u}}$ is any vector in \mathbb{R}^N and not necessarily the S-DEIM reconstruction. Later we will apply this definition with the special case where $\tilde{\mathbf{u}}$ is the S-DEIM reconstruction of the state \mathbf{u} .

2. Unlike the Lipschitz constant, the one-sided Lipschitz constant ρ_f is not necessarily positive [20]. In fact, it can be zero or even a negative number. For instance, consider the one-dimensional function $f(x) = e^{-x} : (-\infty, 0] \rightarrow [1, \infty)$. Since $e^{-x} \geq 1$ on its domain $(-\infty, 0]$, it is easy to see that

$$(x - y)(e^{-x} - e^{-y}) \leq -(x - y)^2 \quad \forall x, y \in (-\infty, 0],$$

and therefore, $\rho_f = -1$. If e^{-x} is considered on the entire real line, then we have $\rho_f = 0$.

3. Any Lipschitz function is also one-sided Lipschitz [20], but the converse is not necessarily the case. For instance, the exponential function e^{-x} is not Lipschitz continuous, but it is one-sided Lipschitz as shown above.

As in Lipschitz continuous functions, the one-sided Lipschitz constant is not unique. From now on, we consider the best, i.e., the smallest possible, value of this constant. In other words, we define the one-sided Lipschitz constant as

$$(4.9) \quad \rho_f = \sup_{\tilde{\mathbf{u}} \neq \mathbf{u}} \frac{\langle \tilde{\mathbf{u}} - \mathbf{u}, \mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u}) \rangle}{\|\tilde{\mathbf{u}} - \mathbf{u}\|^2}.$$

The following result shows that, if \mathbf{f} is Lipschitz continuous, both \mathbf{f} and any orthogonal projection of it are one-sided Lipschitz continuous.

LEMMA 5. Assume $\mathbf{f} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is Lipschitz continuous with a constant $L_f > 0$ and define $\mathbf{g}(\mathbf{u}) = \mathbb{P}\mathbf{f}(\mathbf{u})$, for an arbitrary orthogonal projection \mathbb{P} . Then \mathbf{g} is one-sided Lipschitz continuous with a constant $\rho_g \leq L_f$.

Proof. If \mathbb{P} is the zero matrix, the result holds trivially with $\rho_g = 0$. We now consider nonzero orthogonal projections. We have

$$(4.10) \quad \begin{aligned} \rho_g &= \sup_{\tilde{\mathbf{u}} \neq \mathbf{u}} \frac{\langle \tilde{\mathbf{u}} - \mathbf{u}, \mathbb{P}[\mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u})] \rangle}{\|\tilde{\mathbf{u}} - \mathbf{u}\|^2} \leq \sup_{\tilde{\mathbf{u}} \neq \mathbf{u}} \frac{\|\tilde{\mathbf{u}} - \mathbf{u}\| \cdot \|\mathbb{P}\|_2 \cdot \|\mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u})\|}{\|\tilde{\mathbf{u}} - \mathbf{u}\|^2} \\ &\leq \|\mathbb{P}\|_2 \sup_{\tilde{\mathbf{u}} \neq \mathbf{u}} \frac{\|\mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u})\|}{\|\tilde{\mathbf{u}} - \mathbf{u}\|} \leq L_f, \end{aligned}$$

where we used the Cauchy–Schwartz inequality and $\|\mathbb{P}\|_2$ denotes the spectral norm of \mathbb{P} . Since \mathbb{P} is an orthogonal projection, we have $\|\mathbb{P}\|_2 = 1$. \square

The next theorem is our main convergence result which uses Lemma 5 with the special choice of $\mathbb{P} = \Phi Z Z^\top \Phi^\top$ for the orthogonal projection.

THEOREM 4 (convergence of DAS-DEIM). Consider the system (4.1) and the orthogonal projection $\mathbb{P} = \Phi Z Z^\top \Phi^\top$. Assume that $\mathcal{A} \subset \mathcal{R}[\Phi]$ and that the one-sided Lipschitz constant ρ_g of the vector field $\mathbf{g} := \mathbb{P}\mathbf{f}$ is negative, $\rho_g < 0$.

If $\mathbf{u}_0 \in \mathcal{A}$, then

$$(4.11) \quad \lim_{t \rightarrow \infty} \|\tilde{\mathbf{u}}(t) - \mathbf{u}(t)\| = 0,$$

where $\tilde{\mathbf{u}}(t) = \Phi(S^\top \Phi)^+ \mathbf{y}(t) + \Phi Z \boldsymbol{\xi}(t)$ is the S-DEIM reconstruction with $\boldsymbol{\xi}(t)$ satisfying (4.5). Furthermore, the convergence is exponentially fast.

Proof. Since $\mathbf{u}_0 \in \mathcal{A}$ and the attractor is invariant, we have $\mathbf{u}(t) \in \mathcal{A}$ for all $t \geq 0$. Since $\mathcal{A} \subset \mathcal{R}[\Phi]$, using Corollary 2, we have $\mathbf{u}(t) \in \mathcal{R}[\Phi]$, and therefore,

$$(4.12) \quad \mathbf{u}(t) = \Phi(S^\top \Phi)^+ \mathbf{y}(t) + \Phi Z Z^\top \Phi^\top \mathbf{u}(t)$$

for all $t \geq 0$. Taking the time derivative of this equation, we obtain

$$(4.13) \quad \dot{\mathbf{u}} = \Phi(S^\top \Phi)^+ \dot{\mathbf{y}} + \Phi Z Z^\top \Phi^\top \mathbf{f}(\mathbf{u}).$$

Now let us consider the time derivative of the S-DEIM reconstruction $\tilde{\mathbf{u}}$,

$$(4.14) \quad \dot{\tilde{\mathbf{u}}} = \Phi(S^\top \Phi)^+ \dot{\mathbf{y}} + \Phi Z Z^\top \Phi^\top \mathbf{f}(\tilde{\mathbf{u}}),$$

where we used the fact that $\boldsymbol{\xi}(t)$ satisfies (4.5). Subtracting (4.13) and (4.14), we obtain

$$(4.15) \quad \frac{d}{dt}(\tilde{\mathbf{u}} - \mathbf{u}) = \mathbb{P}[\mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u})],$$

where $\mathbb{P} = \Phi Z Z^\top \Phi^\top$ is an orthogonal projection. Taking the inner product with $\tilde{\mathbf{u}} - \mathbf{u}$, we obtain

$$(4.16) \quad \frac{1}{2} \frac{d}{dt} \|\tilde{\mathbf{u}} - \mathbf{u}\|^2 = \langle \tilde{\mathbf{u}} - \mathbf{u}, \mathbb{P}[\mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u})] \rangle \leq \rho_g \|\tilde{\mathbf{u}} - \mathbf{u}\|^2,$$

where we used the one-sided Lipschitz continuity of $\mathbf{g} = \mathbb{P}\mathbf{f}$.

Finally, using Grönwall's inequality, we have

$$(4.17) \quad \|\tilde{\mathbf{u}}(t) - \mathbf{u}(t)\| \leq \|\tilde{\mathbf{u}}(0) - \mathbf{u}(0)\| e^{\rho_g t}.$$

Since the one-sided Lipschitz constant ρ_g is negative, we obtain the desired result that the error tends to zero as $t \rightarrow \infty$. Clearly, this convergence is exponential in time. \square

We conclude this section with a few remarks regarding Theorem 4. As in case of Lipschitz constant, estimating the smallest one-sided Lipschitz constant is difficult in practice. We also note that the conditions of this theorem are sufficient but not necessary. As our numerical results in section 5 indicate, DAS-DEIM converges for complex dynamical systems, where these conditions are not necessarily met.

Finally, we point out that Theorem 4 can be used to inform the choice of the basis matrix Φ and the selection matrix S . Note that the orthogonal projection $\mathbb{P} = \Phi Z Z^\top \Phi^\top$ depends on these matrices. Therefore, in principle, it might be possible to choose these matrices in such a way that the vector field $\mathbf{g} = \mathbb{P}\mathbf{f}$ has a negative one-sided Lipschitz constant.

5. Numerical results. We present two numerical examples using Lorenz63 and Lorenz96 systems. For each system, two trajectories are numerically generated from different initial conditions. The first trajectory is used as training data to extract POD modes $\{\phi_i\}_{i=1}^m$ and to compute the selection matrix S using the QR algorithm (2.14). The second trajectory is used for testing purposes, i.e., gather observational data, solve the kernel ODE (4.5), and form the corresponding DAS-DEIM estimation (4.2). All errors reported in this section correspond to the testing data set. The observations are recorded at time intervals of $\Delta t = 0.2$. To solve the kernel ODE (4.5), piecewise linear interpolation is used to approximate the observations in intermediate time instances. In the DAS-DEIM Algorithm 4.1, we use the initial guess $\boldsymbol{\xi}_0 = \mathbf{0}$.

We reiterate that (Q)-DEIM was originally developed for reduced-order modeling [10, 14]. Here, however, our goal is estimating the full state of the system from its partial sparse observations without any model order reduction.

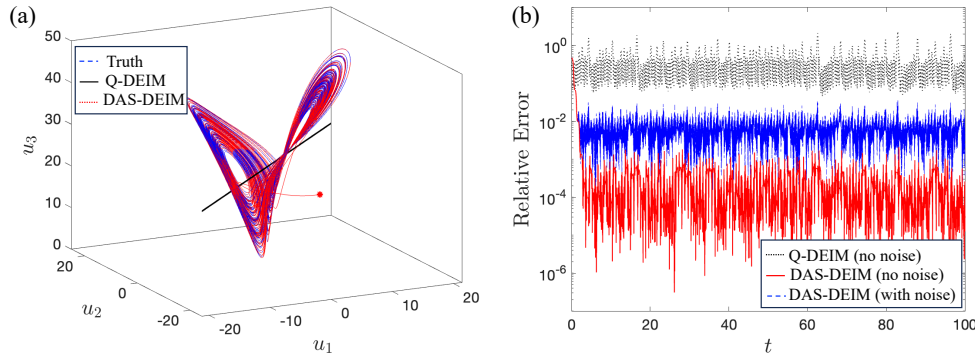


FIG. 3. Vanilla DEIM and S-DEIM state estimation for Lorenz63. (a) State space showing the true trajectory (blue), vanilla Q-DEIM (straight black line), and DAS-DEIM (red). The red circle marks the initial DAS-DEIM estimation. (b) Relative errors as a function of time. (Figure in color online.)

5.1. Lorenz63 system. As a benchmark example, we first consider the Lorenz attractor (or Lorenz63 system),

$$(5.1) \quad \dot{u}_1 = \sigma(u_2 - u_1), \quad \dot{u}_2 = u_1(\rho - u_3) - u_2, \quad \dot{u}_3 = u_1 u_2 - \beta u_3,$$

with the parameters $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$, which is known to exhibit chaotic dynamics. We assume that only $n = 1$ observation is available. The QR algorithm (2.14) determines the second component of the system as most informative; therefore, we set $y(t) = u_2(t)$. We first apply vanilla DEIM with $m = n = 1$ POD mode. As shown in Figure 3(a), since the range of Φ is one-dimensional, the estimated state evolves on a line segment. As a result, the relative error is quite large with a mean of approximately 36%.

For S-DEIM, we can increase the number of modes arbitrarily. Therefore, we use $m = 3$ modes with the same observational time series as before ($y(t) = u_2(t)$). As shown in Figure 3(b), the initial relative error for DAS-DEIM is similar to vanilla DEIM. However, the DAS-DEIM error decreases rapidly as the kernel vector $\mathbf{z}(t)$ converges towards its optimum. Eventually, this error oscillates around 10^{-4} (0.01%), indicating excellent agreement with the true solution of the Lorenz63 system.

We also experiment with noisy data. In particular, we repeat the DAS-DEIM estimation by adding an error $\varepsilon(t) \sim N(0, 0.1)$ to the observations $y(t) = u_2(t)$ at each time instance. Here, $N(0, 0.1)$ denotes the normal distribution with mean zero and standard deviation 0.1. As shown in Figure 3(b), the noise negatively affects the estimation so that the relative error increases, with a mean of 0.7%. Nonetheless, this error is still quite small and significantly smaller than vanilla DEIM error of 36% with clean data.

One may wonder what happens if we use vanilla DEIM with $n = 1$ observations (as before) and $m = 3$ modes. In this case, the vanilla DEIM estimation error increases to 43% (not shown here). This is in line with the numerical results of [12] who observed that the minimum vanilla DEIM error occurs when the number of modes is less than or equal to the number of sensors ($m \leq n$).

5.2. Lorenz96 system. In this section, we consider the Lorenz96 system,

$$(5.2) \quad \dot{u}_i = (u_{i+1} - u_{i-2})u_{i-1} - u_i + F, \quad 1 \leq i \leq N,$$

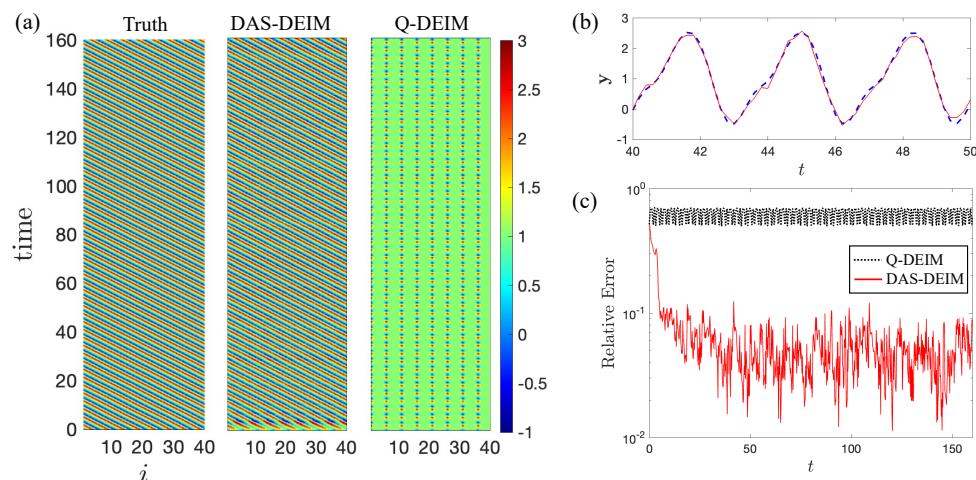


FIG. 4. State estimation for Lorenz96 system. (a) True solution, DAS-DEIM, and vanilla Q-DEIM reconstructions. (b) Close-up view of the observational data from $n = 1$ sensor. Dashed blue shows the true observation data and solid red line marks the corresponding noisy observations. (c) Relative error of vanilla Q-DEIM compared to DAS-DEIM. (Figure in color online.)

with $N = 40$ and periodic boundary conditions: $u_{-1} = u_{N-1}$, $u_0 = u_N$, $u_{N+1} = u_1$. This system is a conceptual model for atmospheric flow and is routinely used for assessing data assimilation methods [9, 28]. Here we use the forcing amplitude of $F = 2$ which leads to a modulating traveling wave solution as shown in Figure 4(a).

The singular values corresponding to the POD modes exhibit a sharp decay after the first five modes: the fifth singular value is around 0.9 whereas the sixth singular value is approximately 4×10^{-8} . This indicates that the attractor \mathcal{A} of the system resides approximately in the linear subspace spanned by the first five POD modes. Therefore, we use $m = 5$ POD modes for both vanilla DEIM and DAS-DEIM estimations. In order to demonstrate that DAS-DEIM performs well even if very few sensors are available, we use $n = 1$ sensor measurements. Column pivoted QR factorization algorithm (2.14) determines u_1 as the best place for sensor placement. Figure 4(b) shows a close-up view of the observational data $y(t) = u_1(t) + \varepsilon(t)$ where random noise $\varepsilon \sim \mathcal{N}(0, 0.1)$ is added to the true data at each time instance. Both vanilla DEIM and DAS-DEIM use this noisy data as input.

As shown in Figure 4(a), vanilla DEIM reproduces incorrect dynamics. In contrast, after initial transients, DAS-DEIM reconstruction converges towards the true solution of the system. Figure 4(c) shows that the vanilla DEIM error oscillates around 62%, whereas DAS-DEIM error decays rapidly and eventually oscillates around 5%. Recall that DAS-DEIM requires the solution of the kernel ODE (4.5) whose dimension $m - n = 4$ is much smaller than the dimension $N = 40$ of the Lorenz96 system. Solving this kernel ODE using MATLAB's ODE45 took only 0.4 seconds.

Finally, we recall Remark 3 which states that the prefactor $\|(S_n^T \Phi_m)^+\|_2$ in the error upper bound (3.17) is a decreasing function of m (the number of modes). Figure 5 demonstrates this numerically with $n = 1$ sensor. As the number of modes m increases, the prefactor $\|(S_n^T \Phi_m)^+\|_2$ decreases indicating that the S-DEIM reconstruction should improve as more modes are used.

6. Conclusions. We introduced Sparse DEIM (S-DEIM) for interpolating unknown functions from their sparse observations when the number of observations is limited. In the special case of continuous-time dynamical systems, we introduced the

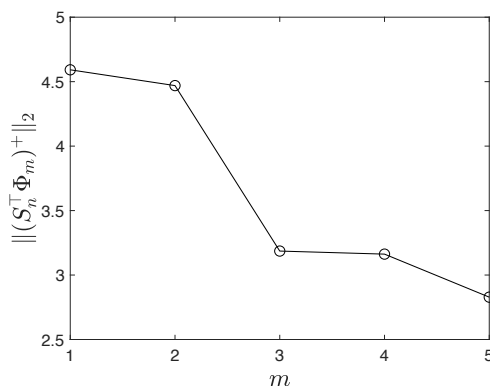


FIG. 5. The matrix norm $\|(S_n^T \Phi_m)^+\|_2$ for the Lorenz96 system with $n = 1$ sensors and increasing number of modes m .

data assimilated S-DEIM (DAS-DEIM) algorithm to efficiently estimate the unknown kernel vector in S-DEIM. Our numerical and theoretical results show great promise for fast and accurate estimation of the state of the system from very few observational data.

A number of open problems remain to be addressed. As we mentioned at the end of section 4.1, Theorem 4 can be used to inform sensor placement. Although here we used the standard column pivoted QR algorithm, a tailor-made sensor placement algorithm for DAS-DEIM should be investigated. Furthermore, Theorem 4 provides sufficient conditions for the convergence of the DAS-DEIM algorithm. These conditions are most likely too restrictive, and therefore, a convergence theorem with necessary and sufficient conditions is desirable.

Finally, we point out that certain data sets, such as those encountered in image processing, are not necessarily generated by a dynamical system. In such cases, alternative methods for approximating the optimal kernel vector of S-DEIM are needed. For instance, deep learning methods can potentially be used. However, our preliminary results (not shown here) indicate that simple neural network architectures (e.g., feed forward) fail to yield an acceptable approximation. In the future, the use of more advanced architectures, such as long short-term memory networks and autoencoders, will be explored.

Acknowledgments. The author is grateful to Professors Ilse Ipsen and Arvind Saibaba for fruitful conversations.

REFERENCES

- [1] A. ASCH, E. J. J. BRADY, H. GALLARDO, J. HOOD, B. CHU, AND M. FARAZMAND, *Model-assisted deep learning of rare extreme events from partial observations*, *Chaos*, 32 (2022), 043112, <https://doi.org/10.1063/5.0077646>.
- [2] P. ASTRID, S. WEILAND, K. WILLCOX, AND T. BACKX, *Missing point estimation in models described by proper orthogonal decomposition*, *IEEE Trans. Automat. Control*, 53 (2008), pp. 2237–2251, <https://doi.org/10.1109/TAC.2008.2006102>.
- [3] A. ATTIA, A. ALEXANDERIAN, AND A. K. SAIBABA, *Goal-oriented optimal design of experiments for large-scale Bayesian linear inverse problems*, *Inverse Problems*, 34 (2018), 095009, <https://doi.org/10.1088/1361-6420/aad210>.
- [4] R. N. BANNISTER, *A review of operational methods of variational and ensemble-variational data assimilation*, *Q. J. Roy. Meteor. Soc.*, 143 (2017), pp. 607–633, <https://doi.org/10.1002/qj.2982>.

- [5] A. BJÖRCK, *Numerical Methods for Least Squares Problems*, Society for Industrial and Applied Mathematics, Philadelphia, 1996, <https://doi.org/10.1137/1.9781611971484>.
- [6] S. L. BRUNTON, B. R. NOACK, AND P. KOUMOUTSAKOS, *Machine learning for fluid mechanics*, *Annu. Rev. Fluid Mech.*, 52 (2020), pp. 477–508, <https://doi.org/10.1146/annurev-fluid-010719-060214>.
- [7] T. BUI-THANH, M. DAMODARAN, AND K. WILLCOX, *Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition*, *AIAA J.*, 42 (2004), pp. 1505–1516, <https://doi.org/10.2514/1.2159>.
- [8] J. L. CALLAHAM, K. MAEDA, AND S. L. BRUNTON, *Robust flow reconstruction from limited measurements via sparse representation*, *Phys. Rev. Fluids*, 4 (2019), 103907, <https://doi.org/10.1103/PhysRevFluids.4.103907>.
- [9] A. CHATTOPADHYAY, P. HASSANZADEH, AND D. SUBRAMANIAN, *Data-driven predictions of a multiscale Lorenz 96 chaotic system using machine-learning methods: Reservoir computing, artificial neural network, and long short-term memory network*, *Nonlinear Proc. Geoph.*, 27 (2020), pp. 373–389, <https://doi.org/10.5194/npg-27-373-2020>.
- [10] S. CHATURANTABUT AND D. C. SORESENSEN, *Nonlinear model reduction via discrete empirical interpolation*, *SIAM J. Sci. Comput.*, 32 (2010), pp. 2737–2764, <https://doi.org/10.1137/090766498>.
- [11] F. L. CHERNOUSKO, *State Estimation for Dynamic Systems*, CRC Press, Moscow, 1993.
- [12] E. CLARK, S. L. BRUNTON, AND J. N. KUTZ, *Multi-fidelity sensor selection: Greedy algorithms to place cheap and expensive sensors with cost constraints*, *IEEE Sens. J.*, 21 (2021), pp. 600–611, <https://doi.org/10.1109/JSEN.2020.3013094>.
- [13] D. DONOHO, *Compressed sensing*, *IEEE Trans. Inform. Theory*, 52 (2006), pp. 1289–1306, <https://doi.org/10.1109/TIT.2006.871582>.
- [14] Z. DRMAČ AND S. GUGERCIN, *A new selection operator for the discrete empirical interpolation method—Improved a priori error bound and extensions*, *SIAM J. Sci. Comput.*, 38 (2016), pp. A631–A648, <https://doi.org/10.1137/15M1019271>.
- [15] Z. DRMAČ AND A. K. SAIBABA, *The discrete empirical interpolation method: Canonical structure and formulation in weighted inner product spaces*, *SIAM J. Matrix Anal. Appl.*, 39 (2018), pp. 1152–1180, <https://doi.org/10.1137/17M1129635>.
- [16] R. EVERSON AND L. SIROVICH, *Karhunen–Loève procedure for gappy data*, *J. Opt. Soc. Am. A*, 12 (1995), pp. 1657–1664, <https://doi.org/10.1364/JOSAA.12.001657>.
- [17] M. FARAZMAND AND A. K. SAIBABA, *Tensor-based flow reconstruction from optimally located sensor measurements*, *J. Fluid Mech.*, 962 (2023), A27, <https://doi.org/10.1017/jfm.2023.269>.
- [18] E. P. HENDRYX, B. M. RIVIÈRE, AND C. G. RUSIN, *An extended DEIM algorithm for subset selection and class identification*, *Mach. Learn.*, 110 (2021), pp. 621–650, <https://doi.org/10.1007/s10994-021-05954-3>.
- [19] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, 1985, <https://doi.org/10.1017/cbo9780511810817>.
- [20] G.-D. HU, *Observers for one-sided Lipschitz non-linear systems*, *IMA J. Math. Control Inform.*, 23 (2006), pp. 395–401, <https://doi.org/10.1093/imamci/dni068>.
- [21] I. C. F. IPSEN AND C. D. MEYER, *The angle between complementary subspaces*, *The American Mathematical Monthly*, 102 (1995), pp. 904–911, <https://doi.org/10.1080/00029890.1995.12004683>.
- [22] S. JI, Y. XUE, AND L. CARIN, *Bayesian compressive sensing*, *IEEE Trans. Signal Process.*, 56 (2008), pp. 2346–2356, <https://doi.org/10.1109/TSP.2007.914345>.
- [23] N. KARNIK, M. G. ABDO, C. E. ESTRADA-PEREZ, J. S. YOO, J. J. COGLIATI, R. S. SKIFTON, P. CALDERONI, S. L. BRUNTON, AND K. MANOHAR, *Constrained optimization of sensor placement for nuclear digital twins*, *IEEE Sens. J.*, 24 (2024), pp. 15501–15516, <https://doi.org/10.1109/JSEN.2024.3368875>.
- [24] G. KIRSTEN, *Multilinear POD-DEIM model reduction for 2D and 3D semilinear systems of differential equations*, *J. Comput. Dyn.*, 9 (2022), 159, <https://doi.org/10.3934/jcd.2021025>.
- [25] A. A. KLISHIN, J. N. KUTZ, AND K. MANOHAR, *Data-Induced Interactions of Sparse Sensors*, preprint, <https://arxiv.org/abs/2307.11838>, 2023.
- [26] J. T. LAUZON, S. W. CHEUNG, Y. SHIN, Y. CHOI, D. M. COPELAND, AND K. HUYNH, *S-opt: A Points Selection Algorithm for Hyper-reduction in Reduced Order Models*, preprint, <https://arxiv.org/abs/2203.16494>, 2022.
- [27] C. L. LAWSON AND R. J. HANSON, *Solving Least Squares Problems*, Society for Industrial and Applied Mathematics, Philadelphia, 1995, <https://doi.org/10.1137/1.9781611971217>.

- [28] R. LGUENSAT, P. TANDEO, P. AILLIOT, M. PULIDO, AND R. FABLET, *The analog data assimilation*, Mon. Weather Rev., 145 (2017), pp. 4093–4107, <https://doi.org/10.1175/MWR-D-16-0441.1>.
- [29] 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 Syst. Mag., 38 (2018), pp. 63–86, <https://doi.org/10.1109/MCS.2018.2810460>.
- [30] S. E. OTTO AND C. W. ROWLEY, *A Discrete Empirical Interpolation Method for Interpretable Immersion and Embedding of Nonlinear Manifolds*, preprint, <https://arxiv.org/abs/1905.07619>, 2019.
- [31] S. E. OTTO AND C. W. ROWLEY, *Inadequacy of linear methods for minimal sensor placement and feature selection in nonlinear systems: A new approach using secants*, J. Nonlinear Sci., 32 (2022), 69, <https://doi.org/10.1007/s00332-022-09806-9>.
- [32] B. PEHERSTORFER, Z. DRMAČ, AND S. GUGERCIN, *Stability of discrete empirical interpolation and gappy proper orthogonal decomposition with randomized and deterministic sampling points*, SIAM J. Sci. Comput., 42 (2020), pp. A2837–A2864, <https://doi.org/10.1137/19M1307391>.
- [33] F. PUKELSHEIM, *Optimal Design of Experiments*, SIAM, Philadelphia, 2006, <https://doi.org/10.1137/1.9780898719109>.
- [34] Y. SHIN AND D. XIU, *Nonadaptive quasi-optimal points selection for least squares linear regression*, SIAM J. Sci. Comput., 38 (2016), pp. A385–A411, <https://doi.org/10.1137/15M1015868>.
- [35] G. STRANG, *The fundamental theorem of linear algebra*, The American Mathematical Monthly, 100 (1993), pp. 848–855, <https://doi.org/10.1080/00029890.1993.11990500>.
- [36] D. B. SZYLD, *The many proofs of an identity on the norm of oblique projections*, Numer. Algorithms, 42 (2006), pp. 309–323, <https://doi.org/10.1007/s11075-006-9046-2>.
- [37] K. WILLCOX, *Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition*, Comput. Fluids, 35 (2006), pp. 208–226, <https://doi.org/10.1016/j.compfluid.2004.11.006>.
- [38] T. A. ZAKI AND M. WANG, *From limited observations to the state of turbulence: Fundamental difficulties of flow reconstruction*, Phys. Rev. Fluids, 6 (2021), <https://doi.org/10.1103/PhysRevFluids.6.100501>.