Data-Driven Approximation of Koopman-Invariant Subspaces with Tunable Accuracy

Masih Haseli Jorge Cortés

Abstract— **This paper studies the problem of identifying finite-dimensional functional spaces that are close (within a predefined level of accuracy) to being invariant under the application of the Koopman operator. Given a dictionary of functions spanning a finite-dimensional functional space and a set of data snapshots gathered from a potentially nonlinear dynamical system, we define a measure of how close a functional space in the span of the dictionary is to being invariant under the Koopman operator. This measure provides a way of determining the prediction accuracy of the functional space. Given a desired level of accuracy, we propose a numerical algorithm, termed Tunable Symmetric Subspace Decomposition (T-SSD), to find a dictionary of functions with elements in the span of the original dictionary that satisfies it. Starting from the original dictionary, the T-SSD algorithm proceeds by iteratively removing the functions that violate the accuracy bound. We prove that T-SSD converges to a dictionary satisfying the accuracy criteria after a finite number of iterations.**

I. Introduction

The Koopman operator associated with a dynamical system characterizes the effect of the dynamics on functions (a.k.a. observables) defined over its state space. The operator is always linear even if the underlying system is nonlinear. This linearity is particularly useful in the context of data-driven identification of dynamical systems because (i) it facilitates the use of efficient linear-algebraic methods suitable for large data sets and (ii) it circumvents the need to assume parametric models capturing the nonlinearity of the dynamics. Despite these advantages, dealing with the infinite-dimensional nature of the Koopman operator requires infinite computational capabilities. This can be addressed by restricting its action to finite-dimensional functional spaces, which in general introduces error in its approximation. This paper focuses on developing data-driven methods to adjust the prediction accuracy of the selected finite-dimensional space.

Literature Review: The eigendecomposition of the Koopman operator, introduced in [1], [2], provides a simple, systematic, and efficient way to represent high-dimensional nonlinear dynamics whose analysis is otherwise cumbersome [3], [4]. This has led to numerous applications, including model reduction [3], control [5]–[8], robotics [9], [10] and network science [11]. Despite its appealing applications, the infinite-dimensional nature of the Koopman operator makes it difficult to employ in data-driven settings. A plausible solution to this issue is to approximate the effect of the operator on a finite-dimensional space. Dynamic Mode Decomposition (DMD) [12] is a popular method for this purpose which was first introduced to extract information from experimental fluid data, but its connection with the Koopman operator became clear later [13]. Subsequently, Extended Dynamic Mode Decomposition (EDMD) was introduced [14] with the main purpose of approximating the effect of the Koopman operator on a finite-dimensional space spanned by a dictionary of functions. EDMD achieves this goal by finding the best linear fit for the evolution of the dictionary based on the available data. Both DMD and EDMD are sensitive to measurement noise [15], [16]. Even with noise-free data, the quality of EDMD's approximation depends on the choice of dictionary. EDMD loses information about the dynamics unless the dictionary spans a space that is invariant under the Koopman operator. This issue has led to investigate the problem of finding Koopman-invariant subspaces [17] with a variety of approaches, including neural networks [18]–[21], sparsity-promoting methods [22], and identification of Koopman eigenfunctions [23], [24]. None of the aforementioned methods provide theoretical results guaranteeing the invariance of the identified subspaces. Our recent works [25], [26] provide algorithms to provably identify all Koopman eigenfunctions and the maximal Koopman-invariant subspace in an arbitrary functional space.

Statement of Contributions: Our goal in this paper is to strike a balance between prediction accuracy and expressiveness when restricting the action of the Koopman operator to finite-dimensional subspaces. This is motivated by the consideration of two opposing forces. On the one hand, it is desirable to consider subspaces as large as possible to be able to capture the evolution of as many observables as possible (including functions containing the states of the system). We refer to this as *expresiveness*. On the other hand, arbitrary subspaces are far from invariant, and this causes large errors in the approximation of the Koopman operator, i.e., in the *prediction accuracy*. Errors in prediction accuracy can be addressed by pruning the subspace to make it invariant at the cost of impacting expresiveness. Our main contribution is the design of a data-driven algorithm to approximate a finite-dimensional functional space that is close to invariant (with tunable accuracy) under the application of the Koopman operator associated with an unknown (potentially nonlinear) dynamical system. We start with a dictionary of functions spanning a finite-dimensional functional space and a set of data snapshots collected from the dynamical system. Building on the introduction of the concept of ϵ-apart subspaces, we provide a parameterized characterization for the gap of a finite-dimensional space

This work was supported by ONR Award N00014-18-1-2828.

Masih Haseli and Jorge Cortés are with Department of Mechanical and Aerospace Engineering, University of California, San Diego, CA 92093, USA, {mhaseli,cortes}@ucsd.edu

from being Koopman invariant based on the available data. Given a desired level of accuracy for the approximation our computational method, termed Tunable Symmetric Subspace Decomposition, iteratively prunes the original dictionary by removing the functions that violate the accuracy bound. We formally establish that the algorithm terminates in a finite number of iterations and its output satisfies the desired accuracy bound. For space reasons, the proofs are omitted here and will appear elsewhere.

II. Preliminaries

Here, we give a brief overview of Koopman operator theory and Extended Dynamic Mode Decomposition (EDMD)[1].

Koopman Operator: Following [4], consider the discrete-time dynamical system defined over the state space $\mathcal{M} \subseteq \mathbb{R}^n$

$$x^+ = T(x). \tag{1}$$

The Koopman operator associated with (1) provides an alternative dynamical description by specifying the evolution of functions, rather than of trajectories. Formally, consider a linear functional space $\mathcal{F}$ defined from state space $\mathcal{M}$ to $\mathbb{C}$ that is closed under composition with T, i.e., $f \circ T \in \mathcal{F}$ for all $f \in \mathcal{F}$. Then, the Koopman operator $\mathcal{K} : \mathcal{F} \to \mathcal{F}$ corresponding to dynamics (1) is defined by

$$\mathcal{K}(f) = f \circ T.$$

Therefore, the Koopman operator maps a function in $\mathcal{F}$ (also known as observable) to a function with temporally shifted values according to the systems trajectories, i.e., given $g = \mathcal{K}(f)$, one can write $g(x) = f(x^+)$ for all $x \in \mathcal{M}$ where $x^+ = T(x)$ is the next temporal point on the trajectory of the system going through x.

Linearity of the functional space $\mathcal{F}$ directly results in the *spatial linearity* of the Koopman operator. Given functions $f_1, f_2 \in \mathcal{F}$ and complex numbers $c_1, c_2 \in \mathbb{C}$, one can write

$$\mathcal{K}(c_1 f_1 + c_2 f_2) = c_1 \mathcal{K}(f_1) + c_2 \mathcal{K}(f_2). \tag{2}$$

[1]Throughout the paper we use the following notations. We denote the set of real, complex, and natural numbers by $\mathbb{R}$, $\mathbb{C}$, and $\mathbb{N}$, respectively. We denote the set of columns, set of rows, number of columns, and number of rows of matrix $A \in \mathbb{C}^{m\times n}$ by $\text{cols}(A)$, $\text{rows}(A)$, $\sharp\text{cols}(A)$, and $\sharp\text{rows}(A)$, respectively. Moreover, A^T, A^H, $A^\dagger$, $\mathcal{R}(A)$, and $\|A\|_F$ represent transpose, conjugate transpose, pseudo-inverse, range space, and Frobenius norm of A. For a square matrix M, we denote its inverse by M^{-1}. Given matrices $A \in \mathbb{C}^{m\times n}$ and $B \in \mathbb{C}^{m\times d}$, we denote by $[A, B] \in \mathbb{C}^{m\times(n+d)}$ the matrix constructed by concatenating A and B side by side. The symbols $\mathbf{0}_{m\times n}$ and I_n represent the m by n zero matrix and the identity matrix of size n, respectively (we drop the indices when the context is clear). For $v \in \mathbb{C}^n$, we represent its 2-norm by $\|v\|_2 := \sqrt{v^H v}$. Given vectors $v_1, \ldots, v_k \in \mathbb{C}^n$, $\text{span}\{v_1, \ldots, v_k\}$ represents the vector space containing all vectors in the form of $c_1 v_1 + \cdots + c_k v_k$ with $c_1, \ldots, c_k \in \mathbb{C}$. Given functions $f_1, \ldots, f_k$, $\text{span}\{f_1, \ldots, f_k\}$ represents the linear functional space containing all functions in the form of $c_1 f_1 + \cdots + c_k f_k$ with $c_1, \ldots, c_k \in \mathbb{C}$. Given the vector space $\mathcal{S} \subseteq \mathbb{R}^m$, we denote by $\mathcal{P}_\mathcal{S}$ the orthogonal projection operator on $\mathcal{S}$. For convenience, given a matrix $A \in \mathbb{R}^{m\times n}$, we denote by $\mathcal{P}_A$ the orthogonal projection operator on $\mathcal{R}(A)$ (we simply refer to orthogonal projections as projections throughout the paper). Given vectors $v, w \in \mathbb{R}^m$, $v \perp w$ means that v and w are orthogonal. Given vector spaces $S_1, S_2 \subseteq \mathbb{R}^m$, $S_1 \perp S_2$ indicates that the subspaces are orthogonal, i.e., all vectors in S_1 are orthogonal to all vectors in S_2. We denote the intersection and union of the sets S_1, S_2 by $S_1 \cap S_2$ and $S_1 \cup S_2$. In addition, we write $S_1 \subseteq S_2$ to indicate that S_1 is a subset of S_2. We denote the composition of functions $f : B \to C$ and $g : A \to B$, by $f \circ g : A \to C$.

The linearity of the Koopman operator allows us to define its eigendecomposition. A function $\phi \in \mathcal{F}$ is called an *eigenfunction* of the Koopman operator with *eigenvalue* λ if

$$\mathcal{K}(\phi) = \lambda\phi. \tag{3}$$

An interesting property of Koopman eigenfunctions is their *linear temporal* evolution. Formally, given the eigenfunction defined in (3), one can write $\phi(x^+) = (\phi \circ T)(x) = \mathcal{K}(\phi)(x) = \lambda\phi(x)$. The combination of linear temporal evolution of the eigenfunctions and spatial linearity of the Koopman operator in (2) enables one to linearly predict the evolution of functions' values on the trajectories of (1). Given eigenfunctions $\{\phi_i\}_{i=1}^{N_k}$ with corresponding eigenvalues $\{\lambda_i\}_{i=1}^{N_k}$, and any function in the form of $f = \sum_{i=1}^{N_k} c_i\phi_i$, one can write

$$f(x(k)) = \sum_{i=1}^{N_k} c_i \lambda_i^k \, \phi_i(x(0)), \quad \forall k \in \mathbb{N}. \tag{4}$$

A subspace $\mathcal{L} \subseteq \mathcal{F}$ is *invariant* under the Koopman operator if for all $f \in \mathcal{L}$ we have $\mathcal{K}(f) \in \mathcal{L}$. For instance, any space generated by Koopman eigenfunctions is invariant.

By making sure that $\mathcal{F}$ contains the states of the system, one can use (4) to fully characterize the dynamics in a linear fashion. However, such functional space $\mathcal{F}$ and the Koopman operator defined on it might be of infinite dimension which complicates the computational analysis of the operator. To circumvent this issue, one can restrict the analysis to finite-dimensional subspaces, as we explain next.

Extended Dynamic Mode Decomposition: Given the computational difficulties emerging from the infinite-dimensional nature of the Koopman operator, Extended Dynamic Mode Decomposition (EDMD) [14] provides a data-driven way to approximate the action of the operator on a finite-dimensional space. Given the dynamical system (1), the EDMD method relies on the following elements:

(i) data matrices $X, Y \in \mathbb{R}^{N\times n}$ comprised of N data snapshots collected from the state space such that

$$y_i = T(x_i), \, \forall i \in \{1, \ldots, N\}, \tag{5a}$$

where x_i^T and y_i^T are the ith rows of X and Y;

(ii) a dictionary $D : \mathcal{M} \to \mathbb{R}^{1\times N_d}$ comprised of N_d functions $\{d_i : \mathcal{M} \to \mathbb{R}\}_{i=1}^{N_d}$ represented by

$$D(x) = [d_1(x), \ldots, d_{N_d}(x)]. \tag{5b}$$

We define the effect of the dictionary on data matrix X as

$$D(X) = [D(x_1)^T, \ldots, D(x_N)^T]^T.$$

The EDMD method approximates the action of the Koopman operator on the functional space spanned by dictionary D by solving the following least-squares problem

$$\underset{K}{\text{minimize}} \|D(Y) - D(X)K\|_F \tag{6}$$

which has the closed-form solution

$$K_{\text{EDMD}} = \text{EDMD}(D, X, Y) := D(X)^\dagger D(Y). \tag{7}$$

K_{EDMD} provides an approximation for the action of the operator on the span of D. Moreover, given the eigendecomposition of K_{EDMD}, one can define the approximated Koopman eigenfunctions. Formally, given $K_{\text{EDMD}}v = \lambda v$ with $\lambda \in \mathbb{C}$ and $v \in \mathbb{C}^{N_d} \setminus \{0\}$, the approximated Koopman eigenfunction ϕ associated with eigenvalue λ is defined as

$$\phi(x) := D(x)v.$$

In general, the restriction to $\text{span}(D)$ by EDMD induces error in the description of the action of the Koopman operator. In fact, the quality of the EDMD approximation depends on the diversity of the data acquired form the system and more importantly on the choice of dictionary D. This motivates the focus on subspaces that are invariant under the Koopman operator. If D spans a Koopman-invariant subspace, the residual error $\|D(Y) - D(X)K_{\text{EDMD}}\|_F$ is equal to zero (independently of the data) and K_{EDMD} recovers perfectly the action of the Koopman operator on $\text{span}(D)$.

III. Problem Statement

We seek to characterize the prediction accuracy of a finite-dimensional functional space (or equivalently, determine how close it is to being invariant under the Koopman operator). The aim is to make use of this characterization to, given any arbitrary functional space, develop computational methods that allow us to identify a subspace with a predetermined desired prediction accuracy. Since in our treatment we assume no knowledge about the underlying dynamics, the aforementioned characterization must be based on the available sampled data from system trajectories.

The elements of our problem statement are the dynamical system (1), data matrices $X, Y \in \mathbb{R}^{N\times n}$ and the dictionary $D : \mathcal{M} \to \mathbb{R}^{1\times N_d}$ in (5). Note that any dictionary $\tilde{D}$ with elements in $\text{span}(D)$ can be characterized by a matrix C by means of the equation $\tilde{D}(x) = D(x)C$. This enables us to utilize conventional efficient linear algebraic methods on dictionary matrices $\tilde{D}(X)$ and $\tilde{D}(Y)$ rather than working with functional spaces directly. Throughout the paper, we rely on the following standard assumption.

Assumption 3.1: (Full Rank Dictionary Matrices): The matrices $D(X)$ and $D(Y)$ have full column rank. □

Assumption 3.1 requires the dictionary matrices to be linearly independent, avoiding redundant functions, and the data snapshots to be diverse and representative of the dynamics. Our main goal is to:

(a) given data matrices X, Y, provide a quantifiable measure to characterize the prediction accuracy over any dictionary $\tilde{D}$ with elements in $\text{span}(D)$;

(b) given a desired level of accuracy, design an algorithm to find a dictionary $\tilde{D}$ with elements in $\text{span}(D)$ that meets the provided accuracy measure.

We deal with (a) in Section IV and with (b) in Section V.

IV. Problem reformulation using ϵ-Apart Spaces

In this section, we take the first step toward developing tunable methods to approximate Koopman eigenfunctions and invariant subspaces by characterizing the quality of subspaces for Koopman approximation. We start by analyzing the quality of approximation derived by the EDMD method. Taking a closer look at (6) reveals that the EDMD solution can be understood as projecting the vectors in $\mathcal{R}(D(Y))$ onto $\mathcal{R}(D(X))$, i.e., for any vector in $v \in \mathcal{R}(D(Y))$ in the form of $D(Y)w$, one can use (7) and write

$$\begin{aligned} D(X)K_{\text{EDMD}}w &= D(X)D(X)^{\dagger}D(Y)w \\ &= D(X)D(X)^{\dagger}v = \mathcal{P}_{D(X)}v, \end{aligned}$$

where in the last equality we have used the fact that $D(X)D(X)^{\dagger}$ is the projection operator on $\mathcal{R}(D(X))$. This projection viewpoint explains the observation that EDMD leads to exactly capturing the Koopman operator's action on an invariant subspace, i.e., if the dictionary D spans a Koopman-invariant subspace and considering Assumption 3.1, $\mathcal{R}(D(X)) = \mathcal{R}(D(Y))$. Hence, in such case, the EDMD projection has perfect prediction accuracy. On the other hand, if $\mathcal{R}(D(Y))$ is orthogonal to $\mathcal{R}(D(X))$,

$$K_{\text{EDMD}} = (D(X)^T D(X))^{-1} D(X)^T D(Y) = \mathbf{0}_{\mathbf{N_d}\times\mathbf{N_d}},$$

where in the first equality we have used Assumption 3.1. This means that the approximation of the Koopman operator by EDMD does not capture any information. The residual error is $\|D(Y) - D(X)K_{\text{EDMD}}\|_F = \|D(Y)\|_F$, i.e., EDMD leads to 100% prediction error on the available data.

The two aforementioned extreme cases ($\mathcal{R}(D(X)) = \mathcal{R}(D(Y))$ and $\mathcal{R}(D(X)) \perp \mathcal{R}(D(Y))$) suggest that one data-driven way to characterize the quality of a dictionary for Koopman approximation is to study the gap between $\mathcal{R}(D(X))$ and $\mathcal{R}(D(Y))$. Motivated by this observation, we define the concept of *ϵ-apart subspaces*.

Definition 4.1: (ϵ-Apart Subspaces): Given $\epsilon \geq 0$, two vector spaces $S_1, S_2 \subseteq \mathbb{R}^p$ are *ϵ-apart* if $\|\mathcal{P}_{S_1}v - \mathcal{P}_{S_2}v\|_2 \leq \epsilon\|v\|_2$, for all $v \in S_1 \cup S_2$. □

Based on Definition 4.1, two vector spaces are ϵ-apart if the norm of difference between a vector v in any of the subspaces and its projection on the other subspace does not exceed $\epsilon\|v\|$. The next result shows that by setting $\epsilon = 0$, one can fully characterize equal subspaces.

Lemma 4.2: (0-apart Spaces are Equal): Vector spaces $S_1, S_2 \subseteq \mathbb{R}^p$ are 0-apart if and only if $S_1 = S_2$. □

Next, we show that all subspaces are 1-apart.

Lemma 4.3: (Any Two Subspaces are 1-apart): Any two vector spaces $S_1, S_2 \subseteq \mathbb{R}^p$ are 1-apart. □

Note that all cases for $\epsilon \geq 1$ are equivalent since for any vector $v \in \mathbb{R}^p$ and any subspace $S \subseteq \mathbb{R}^p$, we always have

$$\|v - \mathcal{P}_S v\|_2 \leq \|v\|_2 \leq \epsilon\|v\|_2, \, \forall \epsilon \geq 1.$$

The previous results enable us to characterize the gap between any two subspaces by the parameter $\epsilon \in [0, 1]$. The case $\epsilon = 0$ implies that the subspaces are identical while in the case $\epsilon = 1$ the subspaces can even be orthogonal. Given the discussion at the beginning of this section, this concept provides a quantifiable measure of prediction accuracy (cf. problem (a) in Section III). Using the concept of ϵ-apart subspaces, we can reformulate problem (b) in Section III as:

Problem 4.4: (ϵ-Apart Subspace Identification): Given $\epsilon \in [0, 1]$, find a dictionary $\tilde{D}$ with elements in $\text{span}(D)$ such that $\mathcal{R}(\tilde{D}(X))$ and $\mathcal{R}(\tilde{D}(Y))$ are ϵ-apart. □

Note that

$$\epsilon^* = \min\{\epsilon \in [0,1] \mid \mathcal{R}(D(X)), \mathcal{R}(D(Y)) \text{ are } \epsilon\text{-apart}\}$$

captures the prediction accuracy of the original dictionary. Therefore, choosing any ϵ in Problem 4.4 smaller than ϵ^* will necessarily result in a smaller dictionary than D, hence reducing expressiveness. It is in this sense that we say that the parameter ϵ captures the trade-off between prediction accuracy and expressiveness of the original dictionary.

V. Tunable Symmetric Subspace Decomposition

In this section, we present a numerical algorithm to solve Problem 4.4 and study its properties.

A. The T-SSD Algorithm

Here, we introduce the Tunable Symmetric Subspace Decomposition (T-SSD) algorithm. An efficient way to tackle Problem 4.4 is to start with the dictionary D and prune it by removing the functions in $\operatorname{span}(D)$ that violate the accuracy bound specified by ϵ. To identify such functions, and inspired by the definition of ϵ-apart spaces, we consider

$$G = \mathcal{P}_{D(X)} - \mathcal{P}_{D(Y)} = D(X)D(X)^\dagger - D(Y)D(Y)^\dagger.$$

The matrix G is symmetric and consequently has real eigenvalues with mutually orthogonal eigenvectors. More importantly, if all eigenvalues of G belong to $[-\epsilon, \epsilon]$, then $\mathcal{R}(D(X))$ and $\mathcal{R}(D(Y))$ are ϵ-apart, and one can conclude that the dictionary does not need pruning. Otherwise, we reduce our attention to the vector space spanned by the eigenvectors of G with eigenvalues in $[-\epsilon, \epsilon]$,

$$\mathcal{W}_\epsilon := \operatorname{span}\{v \in \mathbb{R}^N \mid Gv = \lambda v,\ |\lambda| \leq \epsilon\}.$$

We use this reduced space to search for a dictionary $\tilde{D}$ with the largest dimension such that $\mathcal{R}(\tilde{D}(X)), \mathcal{R}(\tilde{D}(Y)) \subset \mathcal{W}_\epsilon$. Depending on the dimension of $\tilde{D}$ (upon existence), we face two possible scenarios:

(i) $\dim \tilde{D} = \dim D$;
(ii) $\dim \tilde{D} < \dim D$.

Scenario (i) indicates that $\operatorname{span}(\tilde{D}) = \operatorname{span}(D)$ and $\mathcal{R}(D(X)), \mathcal{R}(D(Y)) \subset \mathcal{W}_\epsilon$. Consequently one can deduce that $\mathcal{R}(D(X))$ and $\mathcal{R}(D(Y))$ are ϵ-apart and the dictionary cannot be pruned. Otherwise, in Scenario (ii), we have pruned the dictionary, although there is no guarantee that $\mathcal{R}(\tilde{D}(X))$ and $\mathcal{R}(\tilde{D}(Y))$ are ϵ-apart. To circumvent this issue, we propose to iteratively perform all the aforementioned steps until we cannot prune the dictionary any further.

The aforementioned strategy is formally presented in Tunable Symmetric Subspace Decomposition (T-SSD) algorithm (cf. Algorithm 1). The Symmetric-Intersection routine in Step 8 of the T-SSD algorithm is presented in Algorithm 2[2].

Remark 5.1: (Implementing Algorithm 2 on Finite-Precision Machines): Algorithm 2 depends on calculation of the null space of matrices which is closely related to their rank and can be adversely affected by round-off errors. To address this issue, we use singular value decomposition of the matrix and set small singular values to zero following the method laid out in [25, Remark 5.4]. □

[2]In Algorithms 1-2, the function $\operatorname{basis}(A)$ returns a matrix whose columns form an *orthonormal* basis for $\mathcal{R}(A)$. Also, the function $\operatorname{null}(A)$ returns a matrix whose columns span a basis for the null space of A.

Algorithm 1 Tunable Symmetric Subspace Decomposition

Inputs: $D(X), D(Y) \in \mathbb{R}^{N \times N_d}$, $\epsilon \in [0,1]$

1: **Procedure** T-SSD$(D(X), D(Y), \epsilon)$
2: **Initialization**
3: $i \leftarrow 0$, $A_0 \leftarrow D(X)$, $B_0 \leftarrow D(Y)$, $C_0 \leftarrow I_{N_d}$
4: **while** 1 **do**
5: $\quad i \leftarrow i + 1$
6: $\quad G_i \leftarrow A_{i-1}A_{i-1}^\dagger - B_{i-1}B_{i-1}^\dagger$
$\quad \triangleright$ projection difference
7: $\quad V_i \leftarrow \operatorname{basis}(\{v \in \mathbb{R}^N \mid G_i v = \lambda v,\ |\lambda| \leq \epsilon\})$
$\quad \triangleright$ The eigenpairs corresponding to small eigenvalues
8: $\quad E_i \leftarrow \text{Symmetric-Intersection}(V_i, A_{i-1}, B_{i-1})$
$\quad \triangleright$ Using Algorithm 2
9: $\quad C_i \leftarrow C_{i-1}E_i$ $\quad \triangleright$ Reduce the subspace
10: $\quad A_i \leftarrow A_{i-1}E_i$, $B_i \leftarrow B_{i-1}E_i$
$\quad \triangleright$ Calculating the new dictionary matrices
11: $\quad$ **if** $E_i = 0$ **then**
12: $\quad\quad$ **return** 0
$\quad\quad \triangleright$ The subspace does not exist, returning scalar 0
13: $\quad\quad$ **break**
14: $\quad$ **end if**
15: $\quad$ **if** $\sharp\text{rows}(E_i) \leq \sharp\text{cols}(E_i)$ **then**
16: $\quad\quad$ **return** C_i $\quad \triangleright$ The procedure is complete
17: $\quad\quad$ **break**
18: $\quad$ **end if**
19: **end while**

For convenience, we define the output of the T-SSD algorithm by $C_{\text{T-SSD}} := \text{T-SSD}(D(X), D(Y), \epsilon)$. We use $C_{\text{T-SSD}}$ to build a reduced dictionary $\tilde{D}$ as

$$\tilde{D}(x) := D(x)C_{\text{T-SSD}},\ \forall x \in \mathcal{M}. \tag{8}$$

Moreover, consistently with the EDMD approach, we define the linear prediction matrix $K_{\text{T-SSD}}$ as

$$K_{\text{T-SSD}} = \text{EDMD}(\tilde{D}, X, Y) = \tilde{D}(X)^\dagger \tilde{D}(Y). \tag{9}$$

In addition, given $x \in \mathcal{M}$, one can build $\tilde{D}^+_{\text{pred}}(x)$, the linear predictor for $\tilde{D}(x^+) = \tilde{D} \circ T(x)$, as

$$\tilde{D}^+_{\text{pred}}(x) := \tilde{D}(x)K_{\text{T-SSD}}. \tag{10}$$

Note that every function f in $\operatorname{span}(\tilde{D})$ can be represented by a vector $v \in \mathbb{C}^{\sharp\text{cols}(C_{\text{T-SSD}})}$ in the form of $f = \tilde{D}v$. Accordingly, one can write the predicted value of $f(x^+) = f \circ T(x)$ as

$$f^+_{\text{pred}}(x) := \tilde{D}^+_{\text{pred}}(x)v. \tag{11}$$

Remark 5.2: (Approximating Koopman Eigenfunctions using T-SSD): Note that $K_{\text{T-SSD}}$ is the solution of EDMD applied to the dictionary matrices $\tilde{D}(X)$ and $\tilde{D}(Y)$. As is done in the EDMD approach, we approximate the Koopman eigenfunctions by eigenvectors of $K_{\text{T-SSD}}$. Formally,

Algorithm 2 Symmetric Intersection

Inputs: $V \in \mathbb{R}^{n\times m}$ and $A, B \in \mathbb{R}^{n\times p}$

Procedure Symmetric-Intersection(V, A, B)
if $\text{null}([V, A]) = \emptyset$ **then**
  return 0
  break
else
  $\begin{bmatrix} W_V \\ W_A \end{bmatrix} \leftarrow \text{null}([V, A])$
  ▷ $\sharp\text{cols}(V) = \sharp\text{rows}(W_V), \sharp\text{cols}(A) = \sharp\text{rows}(W_A)$
  if $\text{null}([V, BW_A]) = \emptyset$ **then**
    return 0
    break
  end if
  $\begin{bmatrix} Z_V \\ Z_B \end{bmatrix} \leftarrow \text{null}([V, BW_A])$
  ▷ $\sharp\text{cols}(V) = \sharp\text{rows}(Z_V), \sharp\text{cols}(BW_A) = \sharp\text{rows}(Z_B)$
end if
return $\text{basis}(W_A Z_B)$ ▷ Returning an orthogonal basis

given $K_{\text{T-SSD}}v = \lambda v$, the function $\phi(x) := \tilde{D}(x)v$ is an approximated eigenfunction with corresponding eigenvalue λ. Moreover, based on (10)-(11) and the definition of ϕ,

$$\phi^+_{\text{pred}}(x) := \tilde{D}^+_{\text{pred}}(x)v = \lambda\tilde{D}(x)v = \lambda\phi(x). \qquad \square$$

B. Properties of T-SSD Algorithm

Here, we analyze the T-SSD algorithm. We start by studying the properties of Algorithm 2 (the subroutine of the T-SSD algorithm invoked in Step 8).

Proposition 5.3: (Properties of Algorithm 2): Let matrices V, A, B have full column rank and $E = \text{Symmetric-Intersection}(V, A, B)$ (cf. Algorithm 2). Then,

(a) $E = 0$ or $E^TE = I$;
(b) $\mathcal{R}(AE), \mathcal{R}(BE) \subseteq \mathcal{R}(V)$;
(c) E is maximal, i.e., any nonzero matrix F such that $\mathcal{R}(AF), \mathcal{R}(BF) \subseteq \mathcal{R}(V)$ satisfies $\mathcal{R}(F) \subseteq \mathcal{R}(E)$. $\square$

Now, we are ready to establish the first result regarding the T-SSD algorithm, proving that it always terminates in a finite number of iterations.

Proposition 5.4: (Finite-time Termination of T-SSD): The T-SSD algorithm terminates after at most N_d iterations. $\square$

The next result characterizes important properties of T-SSD's internal matrices.

Lemma 5.5: (Properties of T-SSD Matrices): Let T-SSD algorithm terminate in T time steps. Then,

(a) $\forall i \in \{0, \dots, T-1\}$, $\mathcal{R}(C_{i+1}) \subseteq \mathcal{R}(C_i)$;
(b) $\forall i \in \{0, \dots, T-1\}$, $C_i^TC_i = I$;
(c) $C_T = \mathbf{0}$ or $C_T^TC_T = I$. $\square$

We establish next the main result of the paper, which states that the subspaces spanned by the reduced dictionary obtained from the T-SSD algorithm, cf. (8), applied on available data matrices are ϵ-apart.

Theorem 5.6: (T-SSD Output Subspaces are ϵ-Apart): $\mathcal{R}(\tilde{D}(X))$ and $\mathcal{R}(\tilde{D}(Y))$ are ϵ-apart. $\square$

VI. Simulation Results

We illustrate here the efficacy of the T-SSD algorithm on the Van der Pol oscillator

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= x_2(1 - x_1^2) - x_1, \end{aligned} \tag{12}$$

with $x = [x_1, x_2]^T$. We consider the discretized version of (12) with time step $\Delta t = 0.01s$ and gather data from 2500 trajectories with length equal to two time steps and initial conditions uniformly selected in $[-4, 4]^2$ forming a total of $N = 5000$ data snapshots. To approximate the Koopman operator, we choose the dictionary D constructed with all $N_d = 66$ monomials of degree 10 in the form of $\prod_{i=1}^{10} y_i$, where $y_i \in \{1, x_1, x_2\}$.

Table I shows the dimension of the subspace calculated by T-SSD versus $\epsilon \in \{0.05, 0.15, 0.25, 0.35, 0.45\}$. As expected, as we enforce a tighter accuracy bound (smaller ϵ), the dimension of the subspace identified by T-SSD decreases, showing the trade-off between accuracy and expressiveness. For $\epsilon = 0.45$, the dimension of the T-SSD output is 66, equal to N_d, meaning that the original dictionary satisfies this level of accuracy. Interestingly, the maximal Koopman-invariant subspace in the span of the original dictionary is spanned by the trivial Koopman eigenfunction $\phi(x) = 1$ with eigenvalue $\lambda = 1$, and is therefore one-dimensional. This means that, for this example, focusing on obtaining a subdictionary with a perfect prediction accuracy would lead to not capturing any useful information about the dynamics.

TABLE I: Dimension of subspace identified by T-SSD vs ϵ.

$\boldsymbol{\epsilon}$	0.05	0.15	0.25	0.35	0.45
$\dim \tilde{\boldsymbol{D}}$	8	26	48	56	66

To illustrate the effectiveness of T-SSD in approximating the eigendecomposition of the Koopman operator, we focus on the case with $\epsilon = 0.05$. It is worth mentioning that T-SSD successfully identifies the only eigenfunction of the Koopman operator in the span of the original dictionary ($\phi(x) = 1$ with eigenvalue $\lambda = 1$). It also *approximates* seven more eigenfunctions. Figure 1 illustrates the eigenfunction corresponding to $\lambda = 0.9982 + 0.0078j$, a dominant (closest to the unit circle) oscillatory eigenvalue.

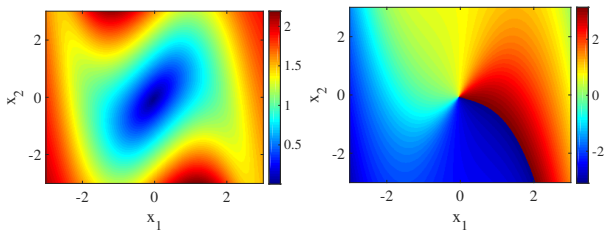

Fig. 1: Absolute value (left) and phase (right) of the eigenfunction corresponding to eigenvalue $\lambda = 0.9982 + 0.0078j$.

To show the benefits of the proposed method regarding prediction accuracy, given a dictionary $\mathcal{D}$ define its EDMD prediction matrix as $K = \text{EDMD}(\mathcal{D}, X, Y)$ where X, Y are the available training data snapshot matrices. We define

the relative prediction error associated to dictionary $\mathcal{D}$ on a trajectory $\{x(k)\}_{k=0}^{M}$ of length M as

$$E_{\text{relative}}(k) := \frac{\|\mathcal{D}(x(k)) - \mathcal{D}(x(0))K^k\|_2}{\|\mathcal{D}(x(k))\|_2} \times 100,$$

where $k \in \{0, \dots, M\}$. Figure 2 compares the one time step relative prediction error of the original dictionary D with the dictionary $\tilde{D}$ identified by T-SSD given $\epsilon = 0.05$. According to the plots, dictionary $\tilde{D}$ has prediction error of 5% or less on most regions of the state space and its prediction error never exceeds 15%. On the other hand, the original dictionary D fails to capture the system's behavior over $[-2, 2]^2$, which is where the limit cycle lies, making it unsuitable for prediction of the Van der Pol oscillator.

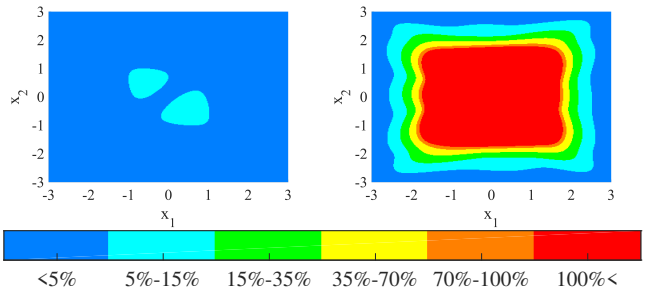

Fig. 2: One step relative error for dictionary identified by T-SSD ($\epsilon = 0.05$) (left) and the original dictionary (right) with initial conditions in $[-3, 3]^2$.

We also investigated the multi-step prediction accuracy of $\tilde{D}$ and D over 500 trajectories with length $M = 10$ time steps with initial conditions uniformly selected in the state space. The average relative error of $\tilde{D}$ on those trajectories did not exceed 2% in one time step and 17% in 10 time steps. On the other hand, the average relative prediction error of the original dictionary D was more than 800% in one time step and 7200% after 10 time steps, underlying the importance of the choice of the dictionary in the accuracy of EDMD.

VII. Conclusions

We have introduced a data-driven approach to identify finite-dimensional functional spaces that have a desired level of prediction accuracy under the application of the Koopman operator. The proposed T-SSD algorithm proceeds by selectively pruning the original dictionary of functions to minimally impact expresiveness while ensuring the desired level of accuracy, and takes a finite number of iterations until convergence. In future work, we aim to provide analytical bounds on storage and computational complexity of the proposed algorithm, investigate ways of enhancing its efficiency, characterize the accuracy of multi-step prediction, and further analyze the theoretical connections of our algorithm with existing methods in the literature.

References

[1] B. O. Koopman, "Hamiltonian systems and transformation in Hilbert space," *Proceedings of the National Academy of Sciences*, vol. 17, no. 5, pp. 315–318, 1931.

[2] B. O. Koopman and J. V. Neumann, "Dynamical systems of continuous spectra," *Proceedings of the National Academy of Sciences*, vol. 18, no. 3, pp. 255–263, 1932.

[3] I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," *Nonlinear Dynamics*, vol. 41, no. 1-3, pp. 309–325, 2005.

[4] M. Budišić, R. Mohr, and I. Mezić, "Applied Koopmanism," *Chaos*, vol. 22, no. 4, p. 047510, 2012.

[5] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[6] S. Peitz and S. Klus, "Koopman operator-based model reduction for switched-system control of PDEs," *Automatica*, vol. 106, pp. 184–191, 2019.

[7] S. H. Son, A. Narasingam, and J. S. Kwon, "Handling plant-model mismatch in Koopman Lyapunov-based model predictive control via offset-free control framework," *arXiv preprint arXiv:2010.07239*, 2020.

[8] Y. Lian, R. Wang, and C. N. Jones, "Koopman based data-driven predictive control," *arXiv preprint arXiv:2102.05122*, 2021.

[9] G. Mamakoukas, M. Castano, X. Tan, and T. Murphey, "Local Koopman operators for data-driven control of robotic systems," in *Robotics: Science and Systems*, Freiburg, Germany, June 2019.

[10] L. Shi and K. Karydis, "Enhancement for robustness of Koopman operator-based data-driven mobile robotic systems," *arXiv preprint arXiv:2103.00812*, 2021.

[11] A. Hasnain, N. Boddupalli, and E. Yeung, "Optimal reporter placement in sparsely measured genetic networks using the Koopman operator," in *IEEE Conf. on Decision and Control*, Nice, France, 2019, pp. 19–24.

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

[13] 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.

[14] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.

[15] S. T. M. Dawson, M. S. Hemati, M. O. Williams, and C. W. Rowley, "Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition," *Experiments in Fluids*, vol. 57, no. 3, p. 42, 2016.

[16] M. Haseli and J. Cortés, "Approximating the Koopman operator using noisy data: noise-resilient extended dynamic mode decomposition," in *American Control Conference*, Philadelphia, PA, July 2019, pp. 5499–5504.

[17] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz, "Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control," *PLOS One*, vol. 11, no. 2, pp. 1–19, 2016.

[18] N. Takeishi, Y. Kawahara, and T. Yairi, "Learning Koopman invariant subspaces for dynamic mode decomposition," in *Conference on Neural Information Processing Systems*, 2017, pp. 1130–1140.

[19] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature Communications*, vol. 9, no. 1, pp. 1–10, 2018.

[20] E. Yeung, S. Kundu, and N. Hodas, "Learning deep neural network representations for Koopman operators of nonlinear dynamical systems," in *American Control Conference*, Philadelphia, PA, July 2019, pp. 4832–4839.

[21] S. E. Otto and C. W. Rowley, "Linearly recurrent autoencoder networks for learning dynamics," *SIAM Journal on Applied Dynamical Systems*, vol. 18, no. 1, pp. 558–593, 2019.

[22] S. Pan, N. Arnold-Medabalimi, and K. Duraisamy, "Sparsity-promoting algorithms for the discovery of informative Koopman invariant subspaces," *arXiv preprint arXiv:2002.10637*, 2020.

[23] M. Korda and I. Mezić, "Optimal construction of Koopman eigenfunctions for prediction and control," 2019. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02278835

[24] E. Kaiser, J. N. Kutz, and S. L. Brunton, "Data-driven discovery of Koopman eigenfunctions for control," *arXiv preprint arXiv:1707.01146*, 2017.

[25] M. Haseli and J. Cortés, "Learning Koopman eigenfunctions and invariant subspaces from data: Symmetric Subspace Decomposition," *https://arxiv.org/abs/1909.01419*, 2020.

[26] ——, "Parallel learning of Koopman eigenfunctions and invariant subspaces for accurate long-term prediction," *https://arxiv.org/abs/2005.06138*, 2020.